UNIVERSIDAD
COMPLUTENSE
MADRID

FACULTAD DE CIENCIAS ECONÓMICAS Y EMPRESARIALES

MÁSTER EN CIENCIAS ACTUARIALES Y FINANCIERAS

TRABAJO DE FIN DE MÁSTER

TÍTULO: *Métodos de aprendizaje automático aplicados a la industria aseguradora*

AUTOR: *Diego Alejandro Ospina Rendón*

TUTORA: *Zuleyka Díaz Martínez*

CURSO ACADÉMICO: *2019-2020*

CONVOCATORIA: *junio 2020*

Índice general

Índice de figuras

Índice de tablas

Introducción

El aprendizaje automático es una rama de la inteligencia artificial que recientemente ha logrado ganar mención en muchos campos a nivel académico y empresarial, dado el amplio espectro de aplicaciones en las cuales tiene utilidad y los innumerables beneficios que puede generar para entender datos y realizar predicciones. Por su parte, la industria aseguradora se encuentra en una etapa de transformación que le exige acogerse a las innovaciones de la era de la información, que entre otras cosas, implica la adopción de nuevas técnicas para saber procesar ese tipo de datos, y así, hacer más eficiente la operatividad y la rentabilidad del negocio. Lo anterior tiene sentido si se considera que las empresas del sector asegurador disponen de una considerable cantidad de información sobre los clientes y su comportamiento. Es más, los actuarios, tradicionalmente ya han usado las matemáticas, la estadística, la economía y los datos, y en la actualidad, emplean cada vez más herramientas informáticas y automatizaciones en sus campos de acción para hacer predicciones y gestionar el riesgo. Por lo anterior, tiene total significado la integración de estos dos campos del conocimiento.

En este trabajo se propone la aplicación de una serie de modelos (análisis de la cesta de compra, análisis clúster, análisis de supervivencia, regresión de Cox, redes neuronales artificiales, árboles de decisión, máquinas de vectores soporte, regularizaciones Ridge, Lasso y Elasticnet, xGBoost tree y xGBoost linear) que se fundamentan en técnicas de aprendizaje automático y de minería de datos con el fin de tener un mejor entendimiento de los clientes, poder sacar un mejor provecho de su relación con la compañía e implementar técnicas modernas para abordar algunas cuestiones de interés dentro del sector asegurador. Dado lo anterior, los modelos propuestos pueden clasificarse en dos categorías: modelos de conocimiento del cliente y modelos actuariales de tarificación. El primer grupo de modelos tiene por objeto caracterizar los perfiles de los clientes, sus patrones de compra y descubrir el comportamiento de la cartera, características reflejadas en cuatro pilares que la compañía debe identificar y aprovechar para tener relaciones más rentables con los clientes: la identificación, la atracción, el desarrollo y la retención. El ciclo de vida de un cliente está enmarcado por la gestión de las relaciones con los clientes, donde las primeras fases consisten en conseguir clientes, posteriormente estas relaciones se establecen y se desarrollan a través de venta cruzada y ventas de mayor valor, y en las últimas fases del ciclo, se procura retenerlos. Por su parte, el segundo grupo de modelos tiene por objeto abordar de una forma diferente el problema de tarificación, a través de técnicas de reciente publicación que pueden mejorar la eficiencia y la calidad de los resultados en comparación con las técnicas tradicionales de GLM, las cuales, a través de los años y la investigación de diferentes autores, se revela que tienen aspectos que pueden mejorar. El objetivo principal de este trabajo consiste en mostrar cómo las técnicas de aprendizaje automático pueden aplicarse en el sector asegurador, usarse en la predicción del comportamiento de los clientes, en la eficiencia del cálculo de primas y traer consigo ventajas para la compañía en términos de estrategia y rentabilidad.

La primera parte de este trabajo presenta los conceptos teóricos necesarios de cada uno de los

doce modelos de aprendizaje automático aplicados. La segunda parte de este trabajo explica el conjunto de datos proveniente de una empresa aseguradora, además, expone los detalles metodológicos y consideraciones técnicas que soportaron el desarrollo de cada uno de los modelos. La tercera parte de este trabajo expone los resultados obtenidos y los análisis respectivos, siempre haciendo mención de la utilidad que cada uno de estos modelos puede traer a la industria aseguradora, para así poner de manifiesto las ventajas de usar estas técnicas de aprendizaje automático en contexto del negocio. Por último, en los anexos se encuentra un detalle más profundo sobre los datos, los paquetes del software y el código utilizado para desarrollar este trabajo.

Capítulo 1

Marco Teórico

En este Capítulo se exponen los conceptos sobre los cuales se basa este trabajo. Por un lado, se describe el concepto de aprendizaje automático y demás relacionados, los cuales constituyen el marco de trabajo para todas las técnicas empleadas. Y por otra parte, se presentan las descripciones técnicas y formulaciones matemáticas de los modelos utilizados.

1.1. Aprendizaje automático

El aprendizaje automático (*machine learning*) es un campo de la inteligencia artificial y de las ciencias de la computación que tiene como objetivo desarrollar técnicas a partir de las cuales los computadores puedan aprender a identificar patrones, anotando que el aprendizaje se alcanza cuando el desempeño mejora con la experiencia; en este contexto, la experiencia está basada en la generación de algoritmos a partir de la generalización y la inferencia de datos e información y no en la programación realizada por un humano. Para ser analizados, los datos pueden estar almacenados en una bodega de datos o en una base de datos (datos estructurados) o también pueden ser extraídos de varias fuentes de datos no estructurados, como por ejemplo archivos, imágenes y audios (Russell y Norvig, 2016). La minería de datos y el aprendizaje automático son conceptos muy similares, con la diferencia básica de que la minería de datos requiere intervención humana para analizar los datos, mientras que en el aprendizaje automático la máquina aprende de forma independiente a partir de esos datos. Los modelos de aprendizaje automático pueden agruparse en dos categorías:

- Modelos supervisados: la finalidad es predecir un evento o estimar el valor de un atributo numérico continuo. En estos modelos se tienen campos o atributos descriptivos (predictores) y un campo objetivo (respuesta) que está asociado a los predictores a través de una función generada por el modelo. Incluyen modelos de clasificación y modelos de estimación.
- Modelos no supervisados: en estos modelos no existe un campo de respuesta, la finalidad es reconocer patrones y estructuras dentro de los datos, los cuales no están guiados por un atributo objetivo específico. Incluye modelos de clúster, modelos de asociación y modelos de reducción de dimensionalidad.

1.2. Minería de datos

La minería de datos (*data mining*) se define como un proceso que utiliza técnicas matemáticas, estadísticas, de inteligencia artificial y de aprendizaje automático para extraer e identificar información útil y posteriormente, obtener conocimiento de las bases de datos. La minería de datos tiene

como objetivo extraer conocimiento y entendimiento a través del análisis de grandes cantidades de datos utilizando técnicas de modelización sofisticadas. Identifica patrones y predice comportamientos, convierte los datos en conocimiento e información valiosa (Tsiptsis y Chorianopoulos, 2011). Las herramientas de tecnología de la información, las tecnologías avanzadas de Internet y la explosión en los datos de los clientes han mejorado las oportunidades de marketing y ha cambiado la forma en que se gestionan las relaciones entre las organizaciones y sus clientes (Ngai, 2005). Además, como refieren Riquelme, Ruiz y Gilbert (2006), las técnicas de minería de datos tienen aplicación en áreas muy diversas de la actividad humana, como por ejemplo: comercio y banca, medicina y farmacia, seguridad y detección de fraude, tratamiento de información no numérica, astronomía, geología, minería, pesca, ciencias ambientales, ciencias sociales, entre otras.

1.3. Gestión de las Relaciones con los Clientes (CRM)

La gestión de las relaciones con los clientes (*customer relationship management* -CRM-) es el proceso que identifica a los clientes de una compañía, crea conocimiento sobre ellos, construye relaciones y moldea las percepciones que éstos tienen sobre los productos o soluciones que reciben (The Sales Educators, 2006). CRM puede entenderse además como una estrategia integral y un proceso de adquirir, retener y asociarse con los clientes para crear valor superior para la empresa y para el cliente mismo (Parvatiyar y Sheth, 2001). Richards y Jones (2006), así como Biswamohan y Bidhubhusan (2012) exponen una serie de beneficios asociados con las estrategias de CRM: mejor habilidad para identificar clientes rentables, posibilidad de hacer ofertas integradas a través de canales, mejor eficiencia y efectividad de los costes y de la fuerza de ventas, capacidad de generar mensajes de marketing individualizados y más efectivos, capacidad de ofrecer productos y servicios personalizados, incremento de la satisfacción y la lealtad del cliente, mejor eficiencia y efectividad del servicio al cliente y mejoras en el proceso de tarificación. La adquisición de una mejor comprensión de los clientes existentes permite a las empresas interactuar, responder y comunicarse de manera más efectiva para mejorar significativamente tasas de retención y de compra (Richards y Jones, 2006).

1.3.1. CRM en el sector seguros

Las crecientes expectativas de los clientes han obligado al sector seguros a introducir iniciativas cada vez más novedosas respecto a la gestión de las relaciones con el cliente ya que esto ha tenido un serio impacto en la venta de los productos de seguros, mientras que los avances tecnológicos y la reducción de costes de las soluciones tecnológicas han reducido las barreras para adoptar medidas de CRM. Como la rentabilidad del sector asegurador depende principalmente de los servicios que se ofrecen y de satisfacer la demanda del cliente de forma regular, esto sugiere que una buena iniciativa de CRM debe ser una fuerte base del sector asegurador. La mayor carga que enfrenta la industria en este sentido es obtener y mantener clientes. Esto se debe al hecho de que cada vez es más difícil para este sector en particular obtener ganancias mientras se reducen los costes (Matis e Ilies, 2014). Debido al aumento de la competencia, las compañías de seguros deben adaptar sus costes y operar eficientemente para sobrevivir en este nuevo entorno (Kasman y Turgutlu, 2011). Además, un cliente satisfecho se mantiene fiel a la empresa, compra también otros productos de seguros y envía mensajes favorables para la imagen de la compañía y sus productos, presta menos atención a las marcas competidoras y a su publicidad y es menos sensible al precio, lo que implica un menor coste de servicio que los nuevos clientes porque las transacciones ya son una cuestión de rutina (Guillén et al., 2012).

1.3.2. Análisis de deserción

Los clientes se convierten en desertores cuando se trasladan a un competidor o simplemente dejan de adquirir los productos o servicios con la compañía, es decir, es el proceso de rotación de clientes. Esta es una preocupación importante para las empresas con muchos clientes que pueden cambiar fácilmente a otros competidores. Con una gestión de deserción efectiva, una empresa puede determinar qué tipo de clientes tienen más probabilidades de abandonar y cuáles tienen más probabilidades de permanecer leales. La minería de datos se puede usar en el análisis de deserción (*churn analysis*) para realizar dos tareas clave: predecir si un cliente en particular abandonará y cuándo sucederá, y comprender por qué los clientes desertan (Richeldi y Perrucci, 2002). Por otra parte, es importante mencionar que el coste de adquirir un nuevo cliente puede exceder sustancialmente el coste de retener a un cliente existente (Siber, 1997).

1.4. Modelos aplicados en este trabajo

1.4.1. Análisis de la cesta de compra (MBA)

El análisis de la cesta de la compra (*market basket analysis* -MBA-) es una técnica de análisis de datos que busca relaciones entre entidades y objetos que con frecuencia aparecen juntos (Loshin y Reifer, 2013). Es una técnica de minería de datos para derivar la asociación entre conjuntos de datos y la concurrencia de elementos nominales o categóricos. Como datos de entrada para el análisis se tienen datos categóricos de registros de transacciones y el resultado del análisis son reglas de asociación que revelan las afinidades entre cada elemento o conjunto de elementos (Raorane, Kulkarni y Jitkar, 2012).

1.4.1.1. Algoritmo a priori

El algoritmo *a priori*, ampliamente usado en el análisis de la cesta de la compra, descubre los conjuntos de elementos frecuentes de una base de datos muy grande a través de una serie de iteraciones. Se requiere el algoritmo *a priori* para generar conjuntos de elementos candidatos, calcular el soporte y reducir los conjuntos de elementos candidatos a los conjuntos de elementos frecuentes en cada iteración. El algoritmo *a priori*, presentado por Agrawal y Srikant (1994) construye un conjunto de elementos, por ejemplo, *itemset1= {Item A, Item B}*. *A priori* luego utiliza un enfoque de abajo hacia arriba, donde los conjuntos de elementos frecuentes se extienden, un elemento a la vez, y funciona eliminando los conjuntos más grandes como candidatos mirando primero los conjuntos más pequeños y reconociendo que un conjunto grande no puede ser frecuente a menos que todos sus subconjuntos lo sean. El algoritmo termina cuando no se encuentran más extensiones exitosas. Este algoritmo genera un conjunto de reglas de asociación en donde un producto o grupo de productos (*Antecedente*) conlleva a la compra de otro producto (*Consecuente*); cada regla está caracterizada por el *soporte o support* (número de ocurrencias en la base de datos), la *confianza o confidence* (probabilidad condicional de que se cumpla la regla) y la *elevación o lift* (relación de dependencia entre los productos; si es mayor que 1 entonces los productos son complementarios, si es menor que 1 entonces los productos son sustitutivos).

1.4.2. Análisis clúster

El análisis clúster (o análisis de conglomerados) abarca una diversidad de técnicas que tienen como objetivo la búsqueda de grupos en un conjunto de individuos. Todo método de clasificación

parte de un conjunto de elementos singulares que deben ser clasificados en un número reducido de grupos, obtenidos por particiones sucesivas del conjunto original y en los que se respete la estructura relacional de similitud entre grupos (Santana, 1991), pero a su vez manteniendo los grupos distintos unos de otros. Se tienen dos tipos principales de métodos de clústers: jerárquicos y no jerárquicos. Los primeros generan un número creciente de clases anidadas, mientras que en los segundos, se generan grupos independientes. Además, existen diversas formas de medir la distancia entre clústers que producen diferentes agrupaciones, como pueden ser la distancia euclídea, Mahalanobis, Manhattan, Chevyshev, entre otras.

1.4.2.1. Algoritmo k-prototypes

Entre los algoritmos de clustering más utilizados, dada su eficacia y simple aplicación, se encuentran los algoritmos k-means para datos numéricos y k-modes para datos categóricos. El algoritmo k-prototypes (Huang, 1997) combina los dos algoritmos mencionados anteriormente para agrupar datos mixtos que tienen variables de tipo numéricas y de tipo categóricas. Cada clúster es caracterizado por su prototipo que se encuentra en el centro de los elementos que componen el clúster. El algoritmo k-prototypes, a través de la siguiente función objetivo, busca minimizar la suma de cuadrados total (S), siendo k el número de clústers definido previamente y n el número de elementos:

$$\arg\min_{S} \sum_{l=1}^{k} \sum_{i=1}^{n} p_{il} d(x_i, Q_l) \tag{1.1}$$

donde $p_{il} \in \{0, 1\}$ es una variable binaria que indica la pertenencia del elemento x_i al clúster l, Q_l es el prototipo (centro) del clúster l y $d(x_i, Q_l)$ es la medida de distancia definida así:

$$d(x_i, Q_l) = \sum_{r=1}^{m_r} (x_{ir} - q_{lr})^2 + \gamma_l \sum_{c=1}^{m_c} \delta(x_{ic}, q_{lc}) \tag{1.2}$$

donde x_{ir} representa los valores en cada uno de los m_r atributos numéricos, x_{ic} representa los valores en cada uno de los m_c atributos categóricos, q_{lr} es la media del atributo numérico r en el clúster l, q_{lc} es la moda del atributo categórico c en el clúster l, γ_l es el peso para los atributos categóricos en el clúster l y para los atributos categóricos se tiene que $\delta(p, q) = 0$ si $p{=}q$ y $\delta(p, q) = 1$ si $p{\neq}q$.

1.4.3. Análisis de supervivencia

El análisis de supervivencia tiene como objeto de estudio el tiempo de permanencia en un estado determinado inicial hasta la ocurrencia de un evento de interés que ocasiona la salida de ese estado inicial. En el análisis de supervivencia, normalmente la variable temporal se refiere al tiempo de supervivencia ya que ésta denota el tiempo que un individuo ha "sobrevivido" durante un periodo de seguimiento a determinado evento de interés. Dicho evento normalmente puede llamarse también "falla" porque usualmente puede ser la muerte, enfermedad o cualquier otra experiencia negativa del individuo. La mayoría de los análisis de supervivencia consideran un problema analítico clave llamado la *censura de datos*. En esencia, la censura ocurre cuando se tiene cierta información sobre el tiempo de supervivencia del individuo, pero no se conoce con exactitud el tiempo total de supervivencia (Kleinbaum y Klein, 2010). Se pueden tener dos tipos de datos censurados:

- Censurados por la derecha: el evento de interés (transición o salida del estado de supervivencia) aún no ha ocurrido en el momento de la observación, así que el tiempo total de

permanencia entre la entrada y la salida del estado es desconocido.

- Censurados por la izquierda: la fecha de entrada al estado no fue observada, así que el tiempo total de permanencia tampoco es conocido.

1.4.3.1. Función de supervivencia

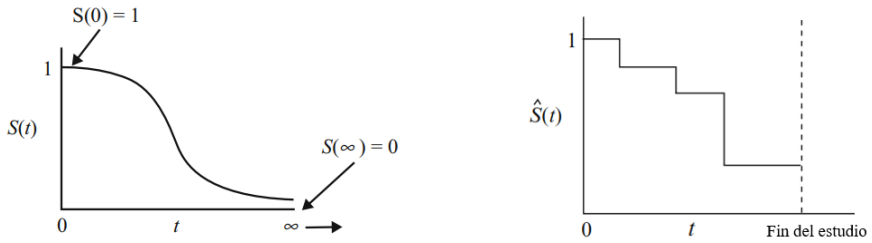
La función de supervivencia *S(t)* se define como la probabilidad de que una persona sobreviva (no le ocurra el evento de interés) más allá que cierto tiempo específico t, es decir, *S(t)* da la probabilidad de que la variable aleatoria T exceda el tiempo específico *t:*

$$S(t) = P(T > t) \tag{1.3}$$

Teóricamente, como t pertenece al intervalo $(0, +\infty)$, la función de supervivencia puede ser graficada como una curva suavizada tal y como se muestra en la Figura 1.1(a). Todas las funciones de supervivencia tienen las siguientes características:

- son decrecientes
- en *t=0*, *S(t)=S(0)=1*; esto es, al inicio del estudio ningún individuo ha sufrido el evento de interés
- en $t=\infty$, $S(t)=S(\infty)=0$; esto es, teóricamente, si el periodo de estudio se prolonga sin límite, eventualmente nadie sobreviviría

Figura 1.1: Funciones de supervivencia



(a) Función de supervivencia $S(t)$ teórica (b) Función de supervivencia $\hat{S(t)}$ real
Fuente: Kleinbaum y Klein (2010).

En la práctica, normalmente se obtienen gráficos de funciones escalonadas, como se muestra en la Figura 1.1(b). Lo anterior se debe a que el periodo de estudio nunca es infinito, a que puede haber riesgos que compitan por la "falla" y a que es probable que no todos los individuos estudiados sufran el evento, por tanto, la función de supervivencia puede no tener tendencia hacia cero al final del estudio.

1.4.3.2. Función de riesgo

La función de riesgo (*hazard function*), denotada por *h(t)*, indica el potencial instantáneo por unidad de tiempo de que el evento ocurra, dado que el individuo ha sobrevivido al tiempo *t*. En algún sentido, la función de riesgo puede ser considerada como el lado opuesto de la información dada por la función de supervivencia:

$$h(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta t | T \geq t)}{\Delta t} \tag{1.4}$$

1.4.3.3. Estimador de Kaplan-Meier

La probabilidad de supervivencia se puede estimar de manera no paramétrica basándose en los tiempos de observación (censurados y no censurados) usando el método de Kaplan-Meier:

$$\hat{S}_{KM}(t) = \prod_{t_i \leq t} \frac{r(t_i) - d(t_i)}{r(t_i)} \tag{1.5}$$

donde $r(t_i)$ es el número de individuos en riesgo y $d(t_i)$ es el número de individuos que sufrieron la ocurrencia del evento de interés en el momento t_i (Kaplan y Meier, 1958).

1.4.4. Regresión de Cox

La regresión de Cox, también conocida como el modelo de riesgo proporcional, es el modelo más utilizado para representar los efectos de un conjunto de variables explicativas sobre la variable del tiempo de supervivencia, o más bien, sobre la probabilidad condicionada de cambio, es decir, sobre la función de riesgo *h(t)* (Pol, 1993):

$$h(t, z) = h_0(t)e^{z\beta} \tag{1.6}$$

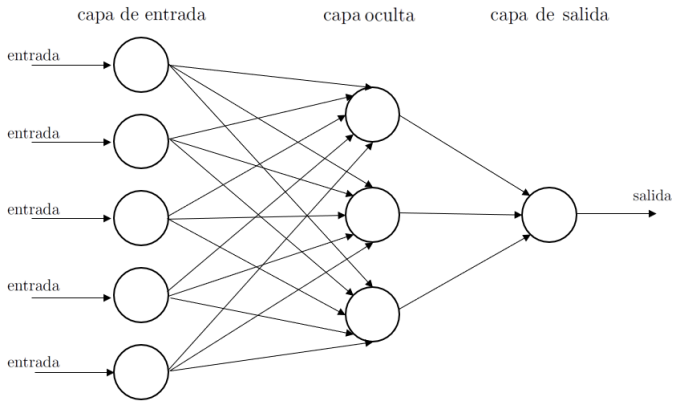
donde z es un vector con los valores de las p variables explicativas para cada individuo, β es un vector $p \times 1$ de parámetros desconocidos y h_0 es una función de riesgo base que corresponde al riesgo de muerte cuando todas las variables explicativas toman el valor de cero, es decir, el conjunto de condiciones estándar $z=0$. Este modelo asume que los riesgos son proporcionales, debido a que el cociente entre el riesgo para dos sujetos con el mismo vector de covariables es constante sobre el tiempo (Cox, 1972):

$$\frac{h(t, z_i)}{h(t, z_j)} = \frac{h_0(t)e^{z_i\beta}}{h_0(t)e^{z_j\beta}} = \frac{e^{z_i\beta}}{e^{z_j\beta}} \tag{1.7}$$

1.4.5. Redes neuronales artificiales

Las redes neuronales artificiales (*artificial neural networks*) constituyen un campo bastante importante dentro de la inteligencia artificial, ya que tratan de crear modelos artificiales que emulan el comportamiento del cerebro humano imitando esquemáticamente la estructura neuronal del cerebro, logrando así un aprendizaje mediante la experiencia y la extracción de conocimiento genérico a partir de un conjunto de datos para crear modelos artificiales utilizados para resolver problemas difíciles de solucionar mediante algoritmos convencionales. En estas redes existen elementos procesadores de información denominados neuronas artificiales, de cuyas interacciones depende el comportamiento del conjunto del sistema (Flórez y Fernández, 2008). Los elementos de procesamiento configuran una o varias capas. Una red neuronal artificial está compuesta normalmente por tres partes: la capa de entrada, constituida por las variables de entrada que reciben la información del entorno; una o varias capas ocultas, las cuales no tienen conexión directa con el entorno; y la capa de salida, que constituye la variable de respuesta de la red neuronal. Los elementos de procesamiento se vinculan a través de fuerzas de conexión variables (ponderaciones o pesos). Esta arquitectura puede verse en la Figura 1.2.

Figura 1.2: Estructura de una red neuronal artificial



Fuente: Elaboración propia.

La red neuronal artificial aprende analizando los registros individuales, creando una predicción para cada uno y ejecutando modificaciones a las ponderaciones si se realiza una predicción errónea, por tanto, la red aprende mediante el entrenamiento, de tal forma que según avanza dicho entrenamiento, la red se hace cada vez más precisa. En forma vectorial, una red neuronal artificial puede representarse así:

$$NET = X * W \tag{1.8}$$

donde NET es la salida, X el vector con los datos de entrada y W el vector de pesos. En la actualidad, la aplicación de las redes neuronales artificiales se ha extendido a distintos campos en los cuales sus características permiten abordar problemas tales como conversión de texto a voz, procesamiento natural del lenguaje, comprensión de imágenes, reconocimiento de patrones, caracteres e imágenes, filtrado de ruido, entre otros (Olabe, 1998). Incluso, Schelldorfer y Wüthrich (2019) presentan cómo pueden combinarse las redes neuronales artificiales con métodos actuariales clásicos de modelos lineales generalizados.

1.4.6. Árboles de decisión

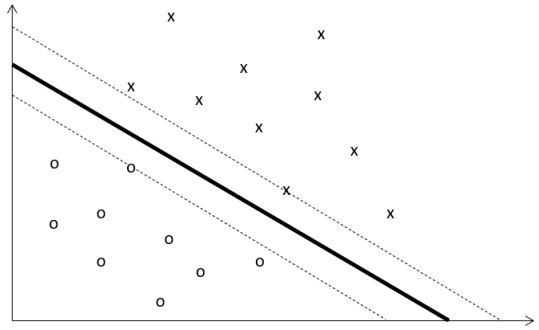
Un árbol de decisión (*decision tree*) es un modelo con estructura de árbol formado por un conjunto de nodos de decisión conectados por ramas extendiéndose hacia abajo desde el nodo raíz hasta terminar en nodos hoja. Comenzando en el nodo raíz, que por convención se coloca en la parte superior del diagrama del árbol de decisión, los atributos o variables se prueban en los nodos de decisión, y cada resultado posible resulta en una rama. Cada rama conduce a otro nodo de decisión o a un nodo hoja de terminación (Larose, 2014). Este modelo es muy fácil de entender y puede ser inducido eficientemente a partir de los datos con los cuales se entrena; además, existen diferentes algoritmos de árboles de decisión para variables objetivo de tipo binarias, categóricas y numéricas, existiendo por tanto, métodos de árboles de clasificación y métodos de árboles de regresión.

1.4.7. Máquinas de vectores soporte (SVM)

Las máquinas de vectores soporte o máquinas de soporte vectorial (*support vector machine* - SVM-) constituyen un método usado tanto para clasificación como para regresión. Las máquinas de vectores soporte para clasificación son clasificadores lineales particulares que se basan en el principio de maximización del margen porque buscan un hiperplano con máxima separación respecto a los

puntos que estén más cerca del mismo, para lograr unas mayor generalización. Es decir, las SVM realizan la tarea de clasificación al construir el hiperplano que separa de manera óptima los datos en categorías en un espacio dimensional superior. Además realizan la minimización del riesgo estructural, es decir, minimizan el error, lo que mejora la complejidad del clasificador, permitiendo lograr un excelente rendimiento de generalización (Adankon y Cheriet, 2007). La Figura 1.3 muestra el hiperplano óptimo que maximiza el margen de separación, diferenciando entre dos tipos de clases.

Figura 1.3: Hiperplano en un espacio de dos dimensiones



Fuente: Adaptado de Cristianini y Shawe-Taylor (2019).

1.4.8. Modelos lineales generalizados (GLM)

Los modelos lineales generalizados (*generalized linear models* -GLM-), muy usados en diferentes campos, son extensiones de los modelos lineales estándar que no necesitan cumplir los supuestos de normalidad y varianza constante. Según explican Goldburd, Khare y Tevet (2016), los GLM son usados para modelizar la relación entre una variable cuyo resultado se desea predecir y una o más variables explicativas. La variable predicha se llama variable objetivo y se denota como y. En aplicaciones de fijación de tarifas de seguros de accidentes, la variable objetivo es típicamente una de las siguientes: recuento de siniestros, severidad de los siniestros, prima pura o ratio de pérdidas. Para variables objetivo de tipo cuantitativas, como las anteriores, los GLM producirán una estimación del valor esperado del resultado. Para otras aplicaciones, la variable objetivo puede ser la ocurrencia o no de un determinado evento, por ejemplo, si un asegurado renovará o no su póliza o si un reclamo presentado contiene fraude. Para tales variables, se puede aplicar un GLM para estimar la probabilidad de que el evento ocurra. Las variables explicativas o predictores se denotan como $x_1, ..., x_p$, donde p es el número de predictores en el modelo. Los predictores potenciales que una aseguradora puede incluir en un plan de tarificación típicamente son las características de la póliza, del titular de la póliza, del objeto asegurado e incluso características geográficas y sociodemográficas.

1.4.8.1. Componentes de un GLM

En un GLM, el resultado de la variable objetivo se compone de un componente sistemático así como de un componente aleatorio. El componente sistemático se refiere a la parte de la variación en los resultados que está relacionada con los valores de los predictores. El componente aleatorio es la parte del resultado impulsado por otras causas diferentes a los predictores del modelo; esto incluye la aleatoriedad pura, es decir, la parte ocasionada por circunstancias impredecibles incluso en teoría, así como lo que puede ser predecible con variables adicionales que no estén en el modelo. En un sentido general, el objetivo al modelizar con GLM es explicar la mayor parte de variabilidad en el resultado que sea posible usando las variables predictoras.

1.4.8.2. El componente aleatorio

En un GLM, la variable objetivo y, se modeliza como una variable aleatoria que sigue cierta distribución de probabilidad. Esa distribución es un miembro de la familia exponencial de distribuciones. La familia exponencial es una clase de distribuciones que tienen ciertas propiedades útiles para trabajarlas mediante GLM. Incluye muchas distribuciones conocidas, como la normal, Poisson, gamma y binomial, entre otras. La selección y especificación de la distribución es una parte importante del proceso de construcción del modelo. La aleatoriedad del resultado de cualquier riesgo particular (denotado y_i) puede ser formalmente expresado de la siguiente manera:

$$y_i \sim Exponencial(\mu_i, \phi) \tag{1.9}$$

Téngase en cuenta que "Exponencial" en la expresión anterior no se refiere a una distribución específica, más bien, es un marcador de posición para cualquier distribución miembro de la familia exponencial. Los términos dentro de los paréntesis se refieren a un rasgo común compartido por todas las distribuciones de la familia: cada miembro toma dos parámetros, μ y ϕ, donde μ es la media de la distribución y ϕ es el parámetro de dispersión, relacionado con la varianza pero no es la varianza. El parámetro μ es de especial interés: como la media de la distribución, representa el valor esperado del resultado, es decir, el resultado del modelo.

1.4.8.3. El componente sistemático

Los GLM modelizan la relación entre la predicción del modelo μ_i y los predictores, como se muestra a continuación:

$$g(\mu_i) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} \tag{1.10}$$

De acuerdo con la Ecuación 1.10, cierta transformación específica de μ_i (denotada $g(\mu_i)$) es igual al intercepto (β_0) más una combinación lineal de los predictores (x_{ij}). Los coeficientes $\{\beta_0, \beta_1, ..., \beta_p\}$ son estimados por el modelo GLM. La transformación de μ_i ($g(.)$) es llamada la función de enlace y es especificada por el usuario. Dicha transformación es de poco interés, realmente el interés se centra en el valor de μ_i, por tanto, es necesario aplicar al resultado la función inversa de $g(.)$. Por otra parte, cuando se usan GLM para la tarificación de seguros, se obtiene un beneficio adicional cuando se especifica que la función de enlace es el logaritmo natural, es decir, $g(x) = \ln x$, ya que se producen estructuras multiplicativas. Los modelos multiplicativos son los tipos más comunes usados en tarificación en el sector asegurador. Por ejemplo, aplicando una función de enlace logarítmica en la Ecuación 1.10, se tiene:

$$\ln \mu_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} \tag{1.11}$$

Y para obtener μ_i, se aplica la función inversa del logaritmo natural a ambos lados de la Ecuación 1.11:

$$\mu_i = e^{\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip}} = e^{\beta_0} \times e^{\beta_1 x_{i1}} \times e^{\beta_2 x_{i2}} \times ... \times e^{\beta_p x_{ip}} \tag{1.12}$$

1.4.9. Regularización

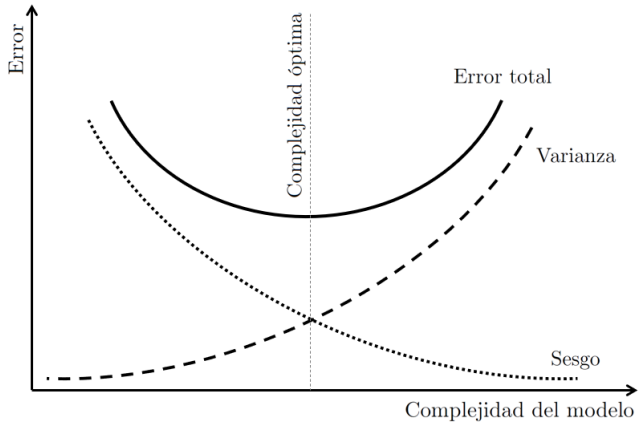
Los modelos de regresión normalmente se ajustan con el método de mínimos cuadrados, pero existen otras alternativas, como la regularización, que traen ventajas al modelo. La regularización es una técnica usada en el campo del aprendizaje automático para reducir el error de un modelo,

ajustando una función apropiada y mejorando tanto la precisión (ajuste) como la interpretabilidad (variables relevantes) del modelo. El ajuste de los parámetros de una regresión se resuelve a través de la minimización de la función de suma de los cuadrados de los residuos (*residual sum of squares* -RSS-):

$$RSS = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \tag{1.13}$$

Un problema que puede surgir es el sobreajuste (*overfitting*), el cual se presenta cuando el modelo obtenido está muy adaptado a un conjunto de datos particular pero no puede generalizarse de forma independiente a otros datos del mismo entorno o a predicciones futuras, reflejando así un mayor ajuste al ruido que a la propia información contenida en los datos (Whiteson et al., 2011). Un problema de interpretabilidad se da cuando las variables usadas en el modelo no están asociadas con la variable respuesta. La regularización implica agregar un término de penalización a la función de error para evitar que los coeficientes alcancen valores grandes, es decir, disminuye la magnitud de los coeficientes o los reduce a cero, disminuyendo significativamente su varianza. Esta penalización permite obtener modelos más simples pero que generalizan mejor. El término de penalización toma varias formas según el método de regularización usado, llevando a una función de error modificada que debe ser minimizada. Una ventaja de la regularización, además de evitar el sobreajuste, consiste en permitir a los modelos complejos ser entrenados con conjuntos de datos de tamaño limitado (Bishop, 2006) a la vez que permite hacer selección de variables. Dado lo anterior, la Figura 1.4 muestra la importancia de la complejidad del modelo, su relación directa con la varianza y su relación inversa con el sesgo.

Figura 1.4: Equilibrio sesgo-varianza



Fuente: Adaptado de Leinweber (1979).

1.4.9.1. Regularización Ridge

Hoerl y Kennard (1970) proponen la técnica Ridge para evitar los problemas de colinealidad en un modelo lineal estimado por mínimos cuadrados. La regresión Ridge es muy similar a la de mínimos cuadrados, con la excepción de que los coeficientes se estiman minimizando una función diferente:

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 = RSS + \lambda \sum_{j=1}^{p} \beta_j^2 \tag{1.14}$$

Donde $\lambda \geq 0$ es el parámetro de penalización y la penalización en este caso es conocida como la norma *L2*. Esta regresión produce un conjunto de soluciones distinto para cada valor de λ y no un único vector de coeficientes estimados como lo hace el método de mínimos cuadrados. El método Ridge tiende a contraer los coeficientes de regresión al incluir el término de penalización en la función objetivo: cuanto mayor sea λ, mayor penalización y por tanto, mayor contracción de los coeficientes. Una forma equivalente de escribir el problema de Ridge es:

$$\hat{\beta}^{ridge} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij} \right)^2 \right\} \quad s.a. \sum_{j=1}^{p} \beta_j^2 \leq t \tag{1.15}$$

donde t es el parámetro de penalización por complejidad. Un inconveniente de este método es que contrae todos los coeficientes hacia cero, pero sin lograr la nulidad exacta de ninguno, por lo que todas las variables son incluidas en el modelo y por tanto, no hay selección de variables.

1.4.9.2. Regularización Lasso

Tibshirani (1996) propone la técnica Lasso (*Least absolute shrinkage and selection operator)*, la cual se asemeja mucho a la regresión Ridge pero con una diferencia en la definición de la penalización, en este caso conocida como la norma *L1*, lo cual trae la ventaja de que algunos coeficientes pueden reducirse hasta exactamente cero, lo cual permite realizar selección de variables a la vez que reduce la varianza del modelo y permite obtener modelos más estables, simples e interpretables. Lasso soluciona el problema de mínimos cuadrados minimizando:

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| = RSS + \lambda \sum_{j=1}^{p} |\beta_j| \tag{1.16}$$

donde $\lambda \geq 0$ es el parámetro de penalización. En forma equivalente, el problema de Lasso es:

$$\hat{\beta}^{lasso} = \arg\min_{\beta} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij} \right)^2 \right\} \quad s.a. \sum_{j=1}^{p} |\beta_j| \leq s \tag{1.17}$$

siendo s el parámetro de penalización por complejidad. Dentro de las limitaciones de Lasso, se tiene que en el caso de que haya más variables que observaciones, Lasso selecciona como máximo un número de variables igual al número de observaciones; además, si se tienen variables con correlaciones muy altas, Lasso selecciona sólo una variable sin ningún criterio específico lo que puede llevar a una pérdida de información y por tanto, a un modelo menos preciso debido a las variables eliminadas. En consecuencia, si se tiene un gran número de variables altamente correlacionadas entre sí, entonces la regresión Ridge presenta mejores resultados ya que ésta contrae todos los coeficientes en forma proporcional sin perder información para el modelo. Además, existe la técnica *group Lasso* propuesta por Yuan y Lin (2006) que permite hacer la selección de grupos de variables, es decir, incluyendo o excluyendo del modelo a la vez todas las variables de un grupo.

1.4.9.3. Regularización ElasticNet

Zou y Hastie (2005) proponen una nueva técnica llamada ElasticNet, la cual conserva las ventajas de Lasso al tiempo que supera algunas de sus limitaciones. Es decir, este método permite seleccionar variables correlacionadas ya que consiste en una combinación convexa de las penalizaciones *L1* y *L2* de los métodos Lasso y Ridge. El problema de optimización en el método ElasticNet es:

$$\hat{\beta}^{ene} = \arg\min_{\beta} |y - X\beta|^2 \quad s.a. \quad \alpha|\beta_1| + (1-\alpha)|\beta_2|^2 \leq t \quad \text{para algún } t \tag{1.18}$$

donde,

$$\alpha = \frac{\lambda_2}{\lambda_1 + \lambda_2}, \quad |\beta_1| = \sum_{j=1}^{p} |\beta_j|, \quad |\beta_2|^2 = \sum_{j=1}^{p} \beta_j^2 \quad \text{y} \quad |y - X\beta|^2 = RSS$$

siendo λ_1 y λ_2 constantes fijas no negativas. Nótese que en este modelo, según la Ecuación 1.18, si $\alpha = 0$, la regresión ElasticNet corresponde a una regresión de Ridge y si $\alpha = 1$, la regresión ElasticNet corresponde a una regresión Lasso. El concepto de grupos de variables de Lasso también puede aplicarse a ElasticNet.

1.4.9.4. Estimación de los parámetros de penalización λ y del parámetro de ajuste ∝

Determinar la cantidad de penalización ($0 < \lambda < \infty$) es extremadamente importante para equilibrar la eliminación de covariables sin importancia mientras se retienen las importantes. Para estimar λ, una posible forma consiste en escoger un conjunto de valores y calcular el error para cada valor y se escoge el λ que haga mínimo dicho error, luego se ajusta el modelo con el valor de λ escogido. Este método también se usa para estimar el parámetro ∝ de ElasticNet. También existen métodos basados en la validación cruzada, en la cual el conjunto de datos se divide de forma aleatoria en k subconjuntos de igual tamaño, entonces uno de los subconjuntos se utiliza como datos de prueba y los demás como datos de entrenamiento; el proceso se repite k veces y finalmente se realiza una media aritmética para obtener un resultado único.

1.4.10. Extreme Gradient Boosting (xGBoost)

Extreme Gradient Boosting (xGBoost) es un algoritmo de aprendizaje automático desarrollado por Chen y Guestrin (2016) basado en árboles de decisión que usa las técnicas de Gradient Boosting, en donde los errores son minimizados por un algoritmo de gradiente descendente y es considerado uno de los mejores algoritmos para trabajar problemas de predicción con datos estructurados. xGBoost presenta la ventaja de que minimiza la velocidad de ejecución y maximiza el rendimiento debido al procesamiento en paralelo, aplicando además, regularizaciones para evitar el sobreajuste del modelo, que puede ser un problema de Gradient Boosting.

Las técnicas de Gradient Boosting (Friedman, 2001) sobre las que se fundamenta xGBoost emplean la lógica en la cual los predictores posteriores aprenden de los errores de los predictores anteriores. Estas técnicas, muy usadas en los problemas de regresión y clasificación, utilizan varios clasificadores débiles, comúnmente los árboles de decisión (en este caso toman el nombre de *Gradient Tree Boosting* -GTB-), pero los resultados se potencian ya que da lugar a un procesamiento secuencial de los datos con una función diferenciable de pérdida, la cual se minimiza tras cada iteración al agregar nuevos modelos, logrando así un pronosticador final más fuerte. Es decir, los nuevos modelos se van adicionando para corregir los errores de los modelos ya existentes (Natekin y Knoll, 2013). En el caso de regresiones, los clasificadores débiles toman el nombre de *Gradient Linear Boosting* -GLB-. Las técnicas de boosting reducen el sesgo y la varianza. Un parámetro importante en esta técnica es la tasa de aprendizaje, que registra el grado de mejora de un árbol respecto al anterior. Una tasa de aprendizaje pequeña significa una mejora lenta pero robusta ante el sobreajuste, lo que se traduce generalmente en mejoras en el resultado a costa de un mayor consumo de recursos de procesamiento.

Capítulo 2

Metodología

En este Capítulo se presenta un detalle de las consideraciones metodológicas que soportaron el diseño de cada modelo empleado; incluyendo las bases de datos usadas, las variables utilizadas y el objetivo perseguido con el uso de cada técnica. Se explica la configuración de los parámetros de los modelos y demás elementos técnicos y estadísticos tenidos en cuenta durante la modelización. Además, se menciona el software y los paquetes del mismo utilizados.

2.1. Datos

En este trabajo se usaron dos bases de datos con información referente a pólizas de clientes asegurados del segmento auto. Un detalle más profundo de estas bases de datos puede consultarse en el Apéndice A.

2.1.1. Transacciones de compra

Estos datos son simulados y contienen dos columnas descritas en la Tabla 2.1. Este conjunto de datos se creó con la finalidad de tener un registro de transacciones de compra artificiales para 1390 pólizas, en las cuales se hace el supuesto que se pueden comprar una serie de paquetes flexibles entre una oferta de 20 complementos disponibles para aumentar los beneficios de la póliza base; cada póliza compra uno o varios complementos según elección del cliente. Estos datos son de aplicación exclusiva para el modelo de análisis de la cesta de compra y un análisis descriptivo de los mismos se presenta en el Apéndice A.1.

Tabla 2.1: Campos del conjunto de datos *Transacciones de compra*

Campo	Descripción
policy	Número de la póliza
product	Complemento adquirido (valores categóricos)

Fuente: Elaboración propia.

2.1.2. European lapse dataset from the direct channel

Estos datos forman parte de una colección de conjuntos de datos que son originales del libro *Computational Actuarial Science with R* editado por Arthur Charpentier (2014), los cuales están disponibles en el paquete *CASdatasets* del software R (Charpentier y Dutang, 2019), junto con otros conjuntos de datos actuariales. En concreto, se usó el *European lapse dataset from the direct channel* (*eudirectlapse*) que contiene información real de una aseguradora anónima de la Unión

Europea y del ramo no vida sobre renovación de 23060 pólizas de seguro de vehículo para un año. El conjunto de datos está compuesto por 19 campos que se describen en la Tabla 2.2, con información sobre los factores de riesgo del tomador de la póliza, de la póliza y del vehículo. Un análisis descriptivo de estos datos se presenta en el Apéndice A.2.

Tabla 2.2: Campos del conjunto de datos *European lapse dataset from the direct channel*

Campo	Descripción
lapse	Variable binaria que indica 0 si el cliente es activo o 1 si el cliente es inactivo
polholder_age	Edad del tomador de la póliza
polholder_BMCevol	Evolución del coeficiente bonus/malus del tomador de la póliza, con tres categorías: "down" si incrementa bonus, "stable" si el coeficiente no cambia y "up" si incrementa malus
polholder_diffdriver	Diferencia de estado entre el tomador de la póliza y el conductor
polholder_gender	Género del tomador de la póliza
polholder_job	Trabajo del tomador de la póliza, con dos categorías: "medical" o "normal"
policy_age	Antigüedad de la póliza
policy_caruse	Uso del vehículo
policy_nbcontract	Número de pólizas del tomador de la póliza con la aseguradora
prem_final	Valor final de renovación de la póliza propuesto al tomador de la póliza
prem_freqperyear	Frecuencia de pago por año de la prima
prem_last	Prima pagada por el tomador de la póliza
prem_market	Prima de mercado
prem_pure	Valor de la prima técnica
vehicl_age	Antigüedad del vehículo
vehicl_agepurchase	Antigüedad del vehículo en el momento de la compra
vehicl_garage	Tipo de garaje
vehicl_powerkw	Caballos de potencia (valores categóricos)
vehicl_region	Región de residencia del tomador de la póliza

Fuente: Adaptado de Charpentier y Dutang (2019).

Adicionalmente, se simularon 2 campos referentes a siniestralidad (número y cuantía de los siniestros), los cuales complementaron dichos datos con el objetivo de poder trabajar todos los modelos con el mismo conjunto de datos ya que éstos ofrecieron la mayor cantidad de campos útiles para los diferentes modelos propuestos en este trabajo, incluyendo campos relevantes para cualquier plan de tarificación (información del tomador, de la póliza y del vehículo). En consecuencia, se decidió realizar este trabajo con la misma base de datos más los campos simulados, primero, para poder enfocarlo como si se tratase de la aplicación a una cartera de clientes o pólizas completa en forma transversal a varios aspectos de la misma y no a varios conjuntos de datos aislados o sin relación, y segundo, porque se pretendió que todos los resultados fueran fáciles de seguir hablando siempre del mismo contexto, cifras y variables para favorecer así una interpretación fluida y conectada entre los diferentes modelos. Para simular el número y la cuantía de los siniestros, se partió de los datos *pg17trainclaim* disponibles en el mismo paquete *CASdatasets*, los cuales corresponden a datos reales de siniestralidad del año 2017 de una empresa privada francesa aseguradora de vehículos, manteniendo así una homogeneidad en el contexto de los datos; luego se procedió a ajustar las distribuciones de esas dos variables a distribuciones Poisson y Gamma respectivamente, y una vez obtenidos los parámetros de media, forma y escala, se simuló el número y la cuantía de los siniestros. Los parámetros usados para la simulación se describen en la Tabla 2.3.

Tabla 2.3: Campos simulados adicionales en *European lapse dataset from the direct channel*

Campo	Descripción	Simulación
claim_nb	Número de siniestros en el año	Poisson($\lambda = 0{,}18$)
claim_amount	Cuantía total de los siniestros en el año	Gamma($k = 1{,}2\,;\theta = 1100$)

Fuente: Elaboración propia.

2.2. Modelización

A través de este trabajo se quiere mostrar cómo diferentes técnicas de aprendizaje automático pueden ayudar a descubrir patrones y predecir comportamientos de los clientes, además de modelizar tarifas para las pólizas, con el objetivo de poder generar un entendimiento estratégico y útil para la rentabilidad de la compañía. En ese orden de ideas, una clasificación y el impacto perseguido por cada uno de los modelos propuestos pueden verse resumidos en la Tabla 2.4. Para los modelos en donde se usa el conjunto de datos *European lapse dataset from the direct channel,* se tiene que, entre las 21 variables que componen dichos datos, se excluyeron para todos los análisis posteriores las variables *prem_final*, *prem_market*, *prem_pure* y *vehicl_agepurchase* debido a las altas correlaciones presentes con otras variables que se mantuvieron, ya que en esencia indican información redundante; lo anterior se detalla en el Apéndice A.2. A continuación, se mencionan los aspectos metodológicos específicos tenidos en cuenta para cada uno de los modelos.

En el *análisis de la cesta de compra* no fue necesario analizar variables relevantes para el modelo puesto que éste utiliza únicamente información transaccional, para lo cual se emplearon los datos artificiales de *Transacciones de compra*. Dentro de este análisis, el algoritmo *a priori* puede generar muchas reglas, sin embargo se limitó el soporte a un mínimo de 0,0025 y la confianza a un mínimo de 0,8 para obtener reglas representativas; además se eliminaron las reglas redundantes, es decir, se excluyeron reglas generales si contenían asociaciones de elementos que a su vez existieran en reglas más particulares.

En el *análisis clúster* se excluyó la variable binaria *lapse* ya que ésta indica la no renovación de la póliza (deserción del cliente), lo cual no se consideró relevante para este análisis, puesto que en otros modelos se trabajará precisamente el tema de la deserción; además, para determinar la cantidad k de clústers se empleó el «método del codo» (*elbow method*), que permite identificar visualmente a través de varias iteraciones (diferentes números de clústers) a partir de qué valor de k se logra una suma de cuadrados racionalmente pequeña.

En el *análisis de supervivencia*, se consideró como evento de interés la cancelación o no renovación de la póliza por parte del asegurado (pérdida del cliente) y como tiempo de vida se considera la antigüedad de la póliza hasta su cancelación, por tanto, se consideraron únicamente las variables *lapse, policy_age* y *polholder_gender;* esta última como variable de estratificación como ejercicio adicional para validar si existía diferencia en el tiempo de vida entre diferentes grupos de clientes, en este caso, el género del tomador de la póliza.

En la *regresión de Cox* se tomó la variable *policy_age* como variable respuesta vinculada con la variable *lapse*, que como se ha dicho anteriormente, indica la ocurrencia del evento de no renovación de la póliza; por otra parte, se incluyeron como variables predictoras *polholder_age*, *polholder_gender*, *polholder_job*, *policy_nbcontract*, *prem_last*, *vehicl_age*, *claim_nb* y *claim_amount*; sin embargo, se validó la significatividad de estas variables iniciales en el modelo de regresión, excluyendo posteriormente las variables no significativas y llegando así a un modelo multivariante con un menor número de variables pero con mayor poder predictivo; a su vez, la significatividad global del modelo se validó con la prueba de *likelihood ratio*, la prueba de Wald y la prueba del

Score logrank; adicionalmente, se validó también el supuesto de proporcionalidad de los riesgos a través de técnicas estadísticas (prueba de hipótesis sobre los residuos escalados de Schoenfeld) y técnicas gráficas (distribución de los residuos en el tiempo).

En los *modelos de clasificación* se propone un modelo de *redes neuronales artificiales*, cuyo objetivo es predecir la deserción del cliente (no renovación de la póliza). Se implementó una arquitectura de perceptrón multicapa (*multi-layer perceptron*) con una capa oculta formada por 80 neuronas artificiales; se usó la variable *lapse* como respuesta del modelo de clasificación y como variables predictoras se usaron *polholder_ age*, *policy_ age*, *policy_ nbcontract*, *prem_ last*, *vehicl_ age*, *claim_ nb* y *claim_ amount*; además, la función de activación usada fue la sigmoide logística. En este modelo se quiso realizar una comparativa entre diferentes métodos adicionales de clasificación mediante aprendizaje automático para poner a prueba la superioridad en precisión lograda a través del enfoque planteado con redes neuronales artificiales. Los métodos alternativos tenidos en cuenta fueron los árboles de decisión y las máquinas de vectores soporte.

En los *modelos de tarificación* se decidió emplear la distribución Tweedie propuesta por Jørgensen y De Souza (1994) para modelizar la prima pura a partir de la variable *claim_ amount*, es decir, las pérdidas esperadas, aunque se conoce que el enfoque tradicional consiste en modelizar por separado la frecuencia y la severidad; sin embargo, la distribución Tweedie también permite obtener resultados razonables ya que ésta no es más que una distribución Poisson compuesta Gamma y permite modelizar distribuciones con asimetría y concentración de masa en cero tal como lo es la distribución de pérdidas totales; además, un supuesto del uso de esta distribución indica que la frecuencia y la severidad de los siniestros se mueven en la misma dirección (Smyth y Jørgensen, 2002), de esta forma, considerando que el coeficiente de correlación de estas dos variables es de 0,678 (Tabla A.4 del Apéndice A.2) se hace viable el uso de esta distribución para el conjunto de datos. Se asumió también que todas las pólizas tienen duración de 1. Las regresiones Tweedie se trabajaron con función de enlace logarítmica, obteniendo así modelos multiplicativos y coeficientes que expresan relatividades respecto a la prima base. Teniendo en cuenta que en estos modelos de tarificación se emplearon algunos métodos que realizan selección de variables, en principio se utilizaron todas las variables predictivas del conjunto de datos para la regresión de la prima pura, permitiendo que cada uno de los métodos realizara la eliminación de variables no significativas. Cabe mencionar además que las variables predictoras numéricas (*polholder_ age*, *policy_ age*, *nbcontract* y *vehicl_ age*) se agruparon en intervalos para convertirlas en variables categóricas y construir así las clases de tarifa (*tariff cells*) como forma de tener en cuenta posibles efectos no lineales de las variables predictoras; dichas discretizaciones se presentan en la Tabla 2.5. Asimismo, para cada variable se asignó como nivel base el nivel con mayor cantidad de observaciones. Dado lo anterior, los diferentes niveles de las variables categóricas aparecen en la matriz de diseño como nuevas variables binarias y por tanto, dentro de las técnicas de regularización se emplearon las técnicas específicas para grupos de variables, es decir, *group Lasso* y *group ElasticNet* para que la selección de variables tuviera sentido. Para la estimación de los parámetros se realizaron validaciones cruzadas empleando 10 subconjuntos en cada iteración. En la última técnica propuesta, xGBoost, se emplearon dos tipos de boosters, uno de tipo lineal (*linear booster*) para poder efectuar la regresión Tweedie y otro de tipo árbol (*tree booster*) que es el convencional de dicha técnica y útil por ejemplo para modelos que no exigen relaciones lineales; y como parámetro de tasa de aprendizaje en esta técnica, se usó $\eta = 0,3$ ($0 \leq \eta \leq 1$), considerando que valores pequeños implican un modelo más robusto al sobreajuste pero requieren un mayor tiempo de computación.

Por último, teniendo en cuenta aspectos técnicos transversales a varios modelos, cabe mencionar que el nivel de significatividad tenido en cuenta para todas las pruebas de hipótesis en este trabajo fue de $\alpha = 0,05$. Además, para los modelos *análisis clúster, redes neuronales artificiales* y *máquinas*

de vectores soporte las variables cuantitativas se normalizaron por el método estándar, haciendo uso de la media y de la desviación estándar de cada variable. Adicionalmente, para mejorar la precisión de estos modelos de clasificación, se realizó un equilibrado de los datos respecto a la variable objetivo *lapse* ya que el nivel lapse=1 representaba originalmente alrededor de un 12 % del total de casos, lo cual generaba predicciones iniciales con muy poca precisión respecto a este nivel objetivo. Y en los modelos en donde fue necesario validar la precisión del aprendizaje automático y de las predicciones, se realizó una partición de los datos originales así: 70 % como conjunto de entrenamiento (*training*) y 30 % como conjunto de prueba (*testing*).

Tabla 2.5: Discretización de variables numéricas

Variable original	Niveles discretizados
polholder_age	[19, 24], (24, 50], (50, 85]
policy_age	[0, 2], (2, 8], (8, 17]
nbcontract	[1, 2], (2, 5], (5, 15]
vehicl_age	[0, 2], (2, 10], (10, 18]

Fuente: Elaboración propia.

2.3. Software

El programa empleado en este trabajo es R (R Core Team, 2019), ya que como lenguaje y entorno para computación estadística, análisis de datos, aprendizaje automático e investigación científica, proporciona una amplia variedad de técnicas estadísticas y de visualización. El Apéndice B ofrece una descripción de los diferentes paquetes de R utilizados en este trabajo. En específico, para la preparación de los datos (normalización, equilibrado y partición) se utilizaron los paquetes *Bbmisc* (Bischl et al., 2017), *DMwR* (Torgo, 2010) y *caret* (Kuhn., 2020); para el *análisis de la cesta de compra* se utilizó el paquete *arules* (Hahsler et al., 2019); para el *análisis clúster* se utilizó el paquete *clustMixType* (Szepannek, 2018); para el *análisis de supervivencia* se utilizaron los paquetes *survminer* (Kassambara, Kosinski y Biecek, 2019) y *muhaz* (Kenneth y Gentleman, 2019); para la *regresión de Cox* se utilizó el paquete *survival* (Therneau, 2020); para los *modelos de clasificación* se utilizaron los paquetes *nnet* (Venables y Ripley, 2002), *rpart* (Therneau y Atkinson, 2019), *rpart.plot* (Milborrow, 2019), *e1071* (Meyer et al., 2019) y *pROC* (Robin et al., 2011); y para los *modelos de tarificación* se utilizaron los paquetes *car* (Fox y Weisberg, 2019), *tweedie* (Dunn, 2017), *HDtweedie* (Wei, Yi y Hui, 2013) y *xgboost* (Chen et al., 2019).

Tabla 2.4: Clasificación e impacto de los modelos propuestos

Categoría	Modelo	Método de aprendizaje automático	Resultado	Impacto
Modelos de conocimiento del cliente/cartera	Análisis de la cesta de compra	Modelo no supervisado	Reglas de asociación de diferentes productos, identificación de patrones	Recomendar acertadamente durante la compra para lograr una mayor venta
	Análisis clúster	Modelo no supervisado	Grupos homogéneos dentro de la cartera de clientes	Caracterizar grupos de clientes/pólizas, segmentar estrategias focalizadas
	Análisis de supervivencia	Modelo supervisado de estimación	Comportamiento del tiempo de vida de las pólizas y probabilidades de supervivencia en cada tiempo t	Identificar cómo se comporta la no renovación de las pólizas respecto al tiempo y entender si existe diferencia entre estratificaciones
	Regresión de Cox	Modelo supervisado de estimación	Modelización del tiempo de vida de las pólizas y características influyentes en el mismo	Entender la relevancia de variables y perfiles que conducen a la no renovación de pólizas
	Redes neuronales artificiales, árboles de decisión y máquinas de vectores soporte	Modelos supervisados de clasificación	Predecir los clientes desertores	Identificar perfiles que llevan a la deserción y evitar la pérdida de estos clientes
Modelos actuariales de tarificación	Regularizaciones Ridge, Lasso y ElasticNet; xGBoost	Modelos supervisados de estimación	Cálculo de primas	Tarificar de manera precisa, eficiente y justa

Fuente: Elaboración propia.

Capítulo 3

Resultados

En este Capítulo se presentan los resultados obtenidos en la aplicación de cada técnica. Se realizan las pruebas estadísticas que sean necesarias para validar los modelos, se analizan e interpretan los resultados más relevantes y se pone en contexto de negocio la utilidad de cada una de estas técnicas, manifestando así las ventajas de su implementación. Y para las técnicas que persiguen el mismo objetivo, se hace un análisis comparativo de sus resultados para tener una visión sobre qué métodos resultaron tener mejor desempeño.

3.1. Análisis de la cesta de compra (MBA)

El análisis de la cesta de compra se implementó a través del algoritmo *a priori* el cual, después de limitar parámetros y eliminar redundancias en las reglas, permitió obtener 19 reglas de asociación las cuales se muestran en la Tabla 3.1. Recuérdese que en este contexto, los productos hacen referencia a complementos de la póliza. A partir de los resultados de la Tabla 3.1 y analizando la Regla 1, se puede concluir que las pólizas que adquieren el Producto N y el Producto Q siempre adquieren también el Producto B (Confianza=1). A partir de la Regla 6, se puede concluir que las pólizas que adquieren los Productos B, C, H y N adquieren el Producto D el 85,7 % de las veces (Confianza=0,857). Dado que todas las reglas tienen elevación superior a 1, se tienen reglas para productos complementarios, es decir, las reglas son aptas para realizar estrategias de venta cruzada de productos y no se tienen productos que compitan entre sí. Así, pueden identificarse qué productos pueden ofrecerse a un tomador de la póliza y, por ejemplo, basándose en la Regla 14 de la Tabla 3.1, se puede ofrecer el Producto B a alguien que compre los Productos H y O, y con un 80 % de probabilidad este cliente aceptará la recomendación de compra.

Tabla 3.1: Reglas de asociación

Regla	Antecedente		Consecuente	Soporte	Confianza	Elevación
1	{Producto N,Producto Q}	=>	{Producto B}	0,00288	1,00000	5,07664
2	{Producto C,Producto Q}	=>	{Producto B}	0,00359	1,00000	5,07664
3	{Producto B,Producto D,Producto Q}	=>	{Producto G}	0,00288	1,00000	8,97419
4	{Producto A,Producto B,Producto S}	=>	{Producto E}	0,00288	1,00000	7,35979
5	{Producto F,Producto I,Producto J}	=>	{Producto A}	0,00288	1,00000	4,44409
6	{Producto B,Producto C,Producto H,Producto N}	=>	{Producto D}	0,00431	0,85714	6,02165
7	{Producto D,Producto H,Producto S}	=>	{Producto J}	0,00359	0,83333	8,71554
8	{Producto B,Producto C,Producto S}	=>	{Producto J}	0,00359	0,83333	8,71554
9	{Producto A,Producto B,Producto F}	=>	{Producto E}	0,00359	0,83333	6,13316
10	{Producto B,Producto G,Producto H}	=>	{Producto D}	0,00719	0,83333	5,85438
11	{Producto E,Producto R}	=>	{Producto D}	0,00288	0,80000	5,62020
12	{Producto G,Producto P}	=>	{Producto D}	0,00288	0,80000	5,62020
13	{Producto G,Producto P}	=>	{Producto B}	0,00288	0,80000	4,06131
14	{Producto H,Producto O}	=>	{Producto B}	0,00288	0,80000	4,06131
15	{Producto E,Producto J,Producto S}	=>	{Producto B}	0,00288	0,80000	4,06131
16	{Producto A,Producto J,Producto L}	=>	{Producto E}	0,00288	0,80000	5,88783
17	{Producto H,Producto J,Producto N}	=>	{Producto C}	0,00288	0,80000	4,46908
18	{Producto B,Producto J,Producto N}	=>	{Producto C}	0,00288	0,80000	4,46908
19	{Producto A,Producto B,Producto G}	=>	{Producto C}	0,00288	0,80000	4,46908

Fuente: Elaboración propia.

Otro análisis relevante consiste en intentar descubrir qué reglas de compra hacen referencia a algún producto en particular. Por ejemplo, puede suponerse que se tiene especial interés en el Producto L, para lo cual se generaron las reglas específicas para ese producto, mostradas en la Tabla 3.2.

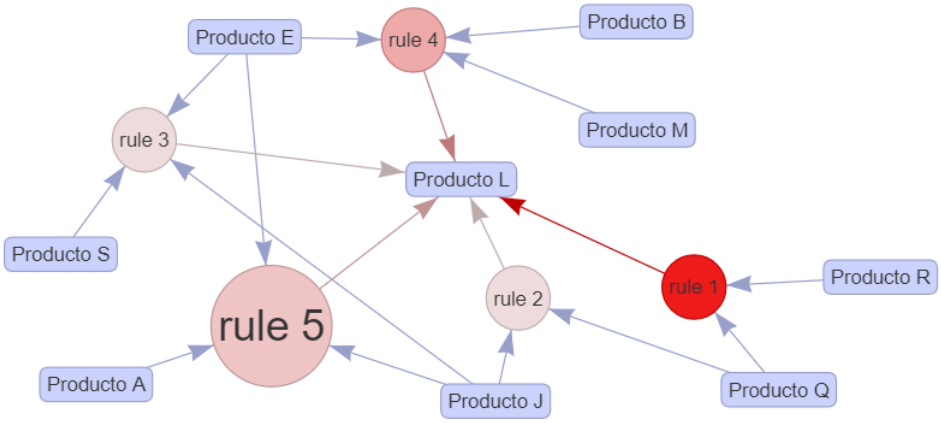
Tabla 3.2: Reglas de asociación para el Producto L

Regla	Antecedente		Consecuente	Soporte	Confianza	Elevación
1	{Producto Q,Producto R}	=>	{Producto L}	0,00216	1,00000	13,63725
2	{Producto J,Producto Q}	=>	{Producto L}	0,00216	0,60000	8,18235
3	{Producto E,Producto J,Producto S}	=>	{Producto L}	0,00216	0,60000	8,18235
4	{Producto B,Producto E,Producto M}	=>	{Producto L}	0,00216	0,75000	10,22794
5	{Producto A,Producto E,Producto J}	=>	{Producto L}	0,00288	0,66667	9,09150

Fuente: Elaboración propia.

Las reglas de asociación para el Producto L también pueden visualizarse en forma de grafos, como lo muestra la Figura 3.1, en donde las flechas están orientadas desde los Antecedentes hacia los Consecuentes. La intensidad del color del círculo de cada regla (*rule*) indica su confianza y el tamaño de dicho círculo indica su soporte.

Figura 3.1: Grafos de las reglas de asociación para el Producto L

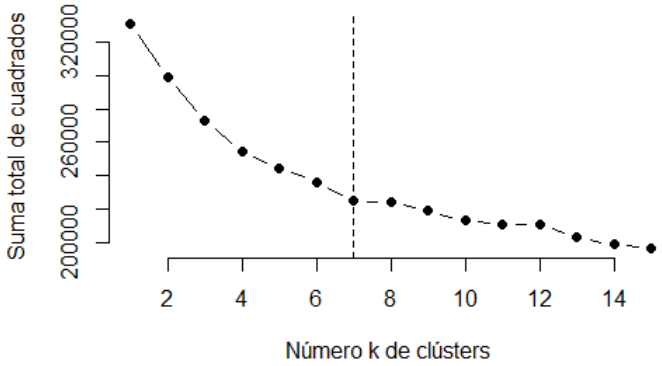


Fuente: Elaboración propia.

3.2. Análisis clúster

El análisis clúster primero tuvo en consideración la normalización de las variables numéricas, ya que el cálculo de las distancias es sensible a la unidad de medida de cada variable. Posteriormente se procedió con la elección del número óptimo de clústers. En la Figura 3.2 puede verse el análisis gráfico del «método del codo», con el cual se decidió realizar 7 clústers, ya que en este valor se logra una disminución significativa en la suma total de cuadrados dentro de los grupos y las disminuciones sucesivas aparentan ser menores o nulas.

Figura 3.2: Selección del número k de clústers



Fuente: Elaboración propia.

Posteriormente se ejecutó el algoritmo k-prototypes. Los resultados de los clústers pueden verse en la Tabla 3.3, en donde se presenta tanto el tamaño de cada grupo (n_i) como los centros (prototipos) para cada variable, recordando que para las variables numéricas el centro corresponde con la media mientras que para las variables categóricas el centro corresponde con la moda. Los clústers con más observaciones son el Clúster 1 (22 % del total) y el Clúster 6 (27 % del total), mientras que por otro lado el Clúster 3 es el que reúne menos observaciones (4 % del total). Por caracterizar algunos de estos clústers, se tiene por ejemplo que el Clúster 1 reúne las pólizas cuyos tomadores tienen las edades más jóvenes (35 años en promedio), principalmente clientes con las mejores calificaciones bonus-malus y las pólizas de menor antigüedad (menores de un año), primas de mayor importe y con pago trimestral; el Clúster 4 reúne principalmente pólizas cuyos tomadores

son mujeres y con vehículos de menor potencia respecto al resto de clústers; el Clúster 5 reúne las pólizas cuyos tomadores tienen las edades más altas (69 años en promedio) y las pólizas de mayor antigüedad (8 años en promedio); por último, el Clúster 7 tiene la particularidad de reunir a los clientes con mayor siniestralidad, ya que presenta las pólizas con mayor número de siniestros y mayores costes de siniestralidad, en contraposición con el Clúster 6 que reúne las pólizas básicamente sin ningún siniestro.

Tabla 3.3: Prototipos para cada clúster generado

Variable	Clústers						
	1	2	3	4	5	6	7
n_i	5052	2436	892	3959	1482	6335	2904
$\%n_i$	22 %	11 %	4 %	17 %	6 %	27 %	13 %
polholder_age	34,97	45,62	44,46	38,94	68,92	45,06	42,50
polholder_BMCevol	down	stable	stable	stable	stable	stable	stable
polholder_diffdriver	same	same	only partner	only partner	same	same	same
polholder_gender	Male	Male	Male	Female	Male	Male	Male
polholder_job	normal	medical	medical	normal	normal	normal	normal
policy_age	0,82	7,73	2,82	1,32	8,28	1,18	2,01
policy_caruse	private or freelance work	unknown	private or freelance work	private or freelance work	unknown	private or freelance work	private or freelance work
policy_nbcontract	1,15	1,29	4,09	1,19	1,22	1,17	1,23
prem_freqperyear	4 per year	1 per year	1 per year	1 per year	1 per year	1 per year	1 per year
prem_last	663,36	319,71	327,02	298,95	290,78	271,53	350,57
vehicl_age	13,12	12,14	13,28	10,21	13,46	14,96	13,20
vehicl_garage	private garage	private garage	private garage	private garage	private garage	private garage	private garage
vehicl_powerkw	75 kW	75 kW	75 kW	25-50 kW	75 kW	75 kW	75 kW
vehicl_region	Reg4	Reg8	Reg7	Reg4	Reg5	Reg4	Reg4
claim_nb	0,06	0,07	0,12	0,03	0,10	0,00	1,12
claim_amount	28,20	37,01	88,52	10,33	72,53	0,52	1545,64

Fuente: Elaboración propia.

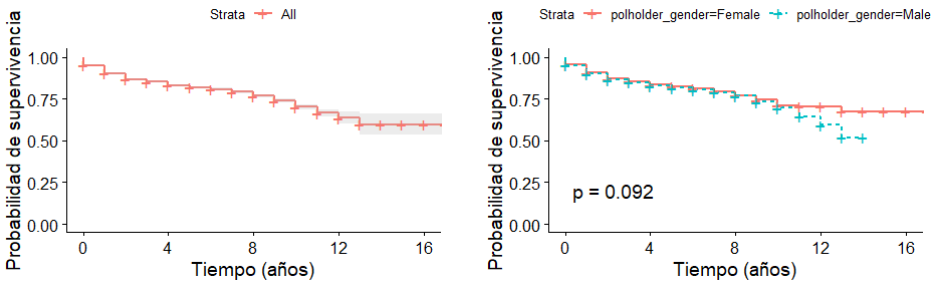
De cara al negocio asegurador, el análisis clúster puede servir para que, una vez identificados grupos de clientes o de pólizas, se creen programas o campañas para clientes específicos, haciendo más eficientes los presupuestos y esfuerzos comerciales; además, sirve también para diagnosticar y entender la composición de la cartera de pólizas, incluso, se pueden asignar nuevos clientes o pólizas a alguno de los grupos ya existentes para enmarcarlo en alguna estrategia ya diseñada.

3.3. Análisis de supervivencia

El análisis de supervivencia se desarrolló teniendo en cuenta como evento de interés la no renovación de la póliza, por tanto, se analiza el tiempo en años que dura activa la póliza hasta su no renovación. Naturalmente se tienen datos censurados ya que el fin del estudio corresponde al periodo de corte de los datos, es decir, sin esperar hasta que la última póliza presente el evento de interés, sin embargo, esto forma parte de las características del modelo. La aplicación del estimador de Kaplan-Meier permitió obtener la función de supervivencia. En la Figura 3.3(a) puede verse la función de supervivencia para todo el conjunto de datos, mientras que en la Figura 3.3(b) puede verse la función de supervivencia estratificada según el género del tomador de la póliza, para identificar si existe diferencia entre un género u otro. Como expresa la Figura 3.3(b), los primeros años de

vida de las pólizas, el comportamiento entre hombres y mujeres es básicamente idéntico, pero a partir de aproximadamente el año 10, las mujeres tienen más probabilidades de supervivencia, esto es, tienen más probabilidades de continuar con su póliza. Sin embargo, para analizar si existe o no diferencia estadísticamente significativa, se realizó la prueba no paramétrica de Mantel-Cox (también conocida como *test logrank*) para contrastar ambas funciones de supervivencia, donde la hipótesis nula consiste en plantear que las curvas de supervivencia de ambas poblaciones no son diferentes. A partir del resultado de la prueba presente en la Figura 3.3(b) (valor p=0,092), no se rechaza la hipótesis nula y se concluye que no hay diferencia significativa en la curva de supervivencia respecto al género del tomador de la póliza.

Figura 3.3: Funciones de supervivencia

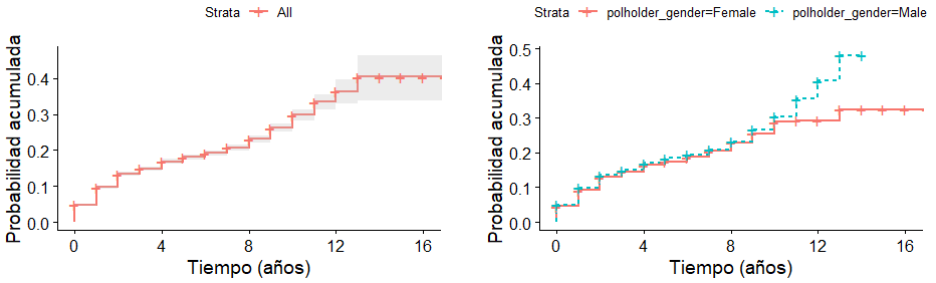


(a) Población total (b) Estratificada por género

Fuente: Elaboración propia.

Por otra parte, la Figura 3.4 muestra las funciones de riesgo acumulado, tanto para la población total como para la estratificada por género. La función de riesgo acumulado se interpreta como la fuerza de mortalidad acumulada.

Figura 3.4: Funciones de riesgo acumulado



(a) Población total (b) Estratificada por género

Fuente: Elaboración propia.

En la Tabla 3.4 se tienen las tablas de supervivencia con las respectivas probabilidades para cada tiempo t y los respectivos límites inferior (LI) y superior (LS) del intervalo de confianza al 95 %. Para la población total, el tiempo medio de supervivencia es de 12,72 años con una desviación estándar de 0,15 (total individuos: 23060, total eventos: 2954); para la población femenina, el tiempo medio de supervivencia es de 12,19 años con una desviación estándar de 0,13 (total individuos: 8339, total eventos: 997) y para la población masculina, el tiempo medio de supervivencia es de 11,54 años con una desviación estándar de 0,21 (total individuos: 14721, total eventos: 1957).

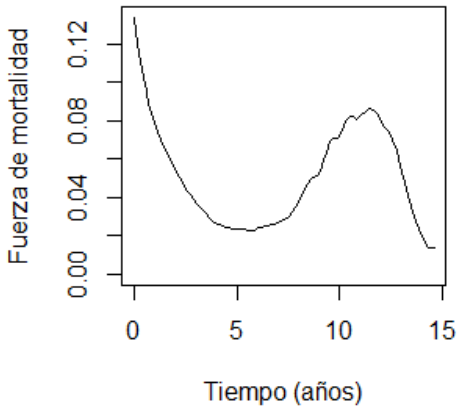
Tabla 3.4: Tablas de supervivencia

Tiempo	Población total			polholder_gender=Female			polholder_gender=Male		
	Probabilidad	LI 95 %	LS 95 %	Probabilidad	LI 95 %	LS 95 %	Probabilidad	LI 95 %	LS 95 %
0	0,952	0,949	0,955	0,954	0,950	0,959	0,952	0,949	0,955
1	0,903	0,899	0,907	0,907	0,900	0,914	0,903	0,899	0,907
2	0,867	0,861	0,872	0,871	0,862	0,880	0,867	0,861	0,872
3	0,852	0,846	0,858	0,855	0,845	0,865	0,852	0,846	0,858
4	0,832	0,825	0,838	0,836	0,824	0,847	0,832	0,825	0,838
5	0,819	0,812	0,826	0,826	0,814	0,838	0,819	0,812	0,826
6	0,808	0,801	0,816	0,813	0,800	0,826	0,808	0,801	0,816
7	0,792	0,784	0,800	0,795	0,781	0,809	0,792	0,784	0,800
8	0,768	0,759	0,778	0,770	0,754	0,787	0,768	0,759	0,778
9	0,738	0,726	0,749	0,745	0,726	0,764	0,738	0,726	0,749
10	0,702	0,687	0,717	0,712	0,688	0,736	0,702	0,687	0,717
11	0,665	0,644	0,686	0,706	0,681	0,733	0,665	0,644	0,686
12	0,637	0,604	0,671	-	-	-	0,637	0,604	0,671
13	0,596	0,535	0,662	0,676	0,615	0,743	0,596	0,535	0,662

Fuente: Elaboración propia.

Por último, la Figura 3.5 presenta la función de la fuerza de mortalidad o razón instantánea de ocurrencia del evento (*hazard rates*) para la población total. Esta describe el riesgo de que un evento ocurra en un instante t, condicionado a que el evento aún no ha ocurrido. Por lo que puede verse, dicha fuerza de mortalidad es decreciente en los primeros años de vida de la póliza, lo cual implica pocas cancelaciones, pero alcanza un máximo alrededor de los 12 años, donde precisamente se tiene la vida promedio de las pólizas.

Figura 3.5: Función de fuerza de mortalidad



Fuente: Elaboración propia.

3.4. Regresión de Cox

El primer paso para realizar la regresión de Cox consistió en comprobar el papel que cada variable explicativa juega por sí sola, para esto, se hizo la regresión de Cox univariante para cada una de las variables predictoras sobre la variable respuesta, en este caso, los años de duración ininterrumpida de la póliza (*policy_ age*). Los resultados se pueden ver en la Tabla 3.5. Es importante hacer notar que la columna *Riesgo (HR) -Hazard Ratio-* indica el efecto multiplicativo del coeficiente β en el modelo de regresión de Cox y es la forma de cuantificar el impacto de cada variable sobre la variable objetivo, de tal forma que si HR=1 no hay efecto de la variable sobre la probabilidad

del evento, si HR<1 hay una reducción en el riesgo (aumenta la probabilidad de supervivencia) y si HR>1 hay un incremento en el riesgo (disminuye la probabilidad de supervivencia). Analizando las diferentes significatividades a partir del valor p de la Tabla 3.5, que permite contrastar la significatividad del coeficiente β de cada variable a través de la hipótesis nula de que éste es igual a cero (o, en forma equivalente, que el coeficiente HR respectivo es igual a uno), por lo que dicha variable no sería significativa y puede eliminarse del modelo. Una forma alternativa de comprobar la significatividad de la variable consiste en revisar si el intervalo de confianza para HR contiene el valor de 1, ya que en este caso la variable aporta muy poco al riesgo. Dado lo anterior, se excluyeron las variables *polholder_gender*, *claim_nb* y *claim_amount*; además, se decidió excluir también la variable *prem_last* porque a pesar de que el valor p es pequeño, el β es bastante cercano a 0 y el HR es muy cercano a 1. Teniendo en cuenta los coeficientes para las variables significativas, se puede concluir que a mayor edad (*polholder_age*), mayor número de contratos (*policy_nbcontract*) y mayor antigüedad del vehículo (*vehicl_age*) se tiene más probabilidad de supervivencia (β negativo, o lo que es lo mismo, HR menor que 1, reduciendo el riesgo), es decir, se tienen pólizas más duraderas (se aumenta la probabilidad de renovación).

Tabla 3.5: Resultados de la regresión de Cox para cada variable por separado

Variable	$\hat{\beta}$	Riesgo (HR)	HR LI 95 %	HR LS 95 %	Valor p
polholder_age	-0,043	0,958	0,954	0,961	1,007E-123
polholder_gender	0,066	1,068	0,990	1,153	0,091
polholder_job	0,175	1,192	1,107	1,283	3,541E-06
policy_nbcontract	-0,085	0,919	0,873	0,967	0,001
prem_last	0,001	1,001	1,001	1,001	1,627E-52
vehicl_age	-0,034	0,966	0,957	0,976	1,250E-11
claim_nb	0,009	1,009	0,927	1,099	0,833
claim_amount	-2,540E-06	1,000	1,000	1,000	0,924

Fuente: Elaboración propia.

El siguiente paso en este modelo consistió en desarrollar la regresión de Cox multivariante para entender cómo los factores en forma conjunta influyen en la supervivencia. Los resultados se presentan en la Tabla 3.6 en donde puede verse que todas las variables empleadas se mantienen significativas.

Tabla 3.6: Resultados de la regresión de Cox multivariante

Variable	$\hat{\beta}$	Riesgo (HR)	HR LI 95 %	HR LS 95 %	Valor p
polholder_age	-0,043	0,958	0,955	0,961	<2E-16
polholder_jobnormal	0,271	1,311	1,217	1,413	1,080E-12
policy_nbcontract	-0,064	0,938	0,891	0,987	0,014
vehicl_age	-0,027	0,973	0,964	0,983	1,150E-07

Fuente: Elaboración propia.

En cuanto a la significatividad global del modelo y según la Tabla 3.7 que muestra las pruebas de likelihood ratio, Wald y del Score logrank, los valores p de cada prueba indican que el modelo también es significativo, ya que permiten rechazar la hipótesis nula de que todos los coeficientes β son 0. Estos métodos son asintóticamente equivalentes y para un número suficientemente grande de observaciones los resultados son similares, como ocurre en este caso. De forma análoga a como ocurre en los modelos univariantes, la edad, el número de contratos y la antigüedad del vehículo reducen el riesgo de no renovación (aumentan la probabilidad de supervivencia). Particularmente para la variable *polholder_job*, al ser una variable categórica con dos niveles (*medical* o *normal*),

ésta queda expresada como *polholder_jobnormal* ya que indica la diferencia en el riesgo del segundo nivel (*normal*) respecto al primer nivel (*medical*), esto es, un empleo de tipo *normal* aumenta el riesgo en un 31,1 % con respecto a un trabajo de tipo *medical*. A manera de interpretación de los coeficientes, se tiene que, conservando todas las demás variables constantes, un aumento de 1 año en la edad del tomador de la póliza reduce el riesgo anual de no renovación en 4 %, un aumento de 1 contrato en los que posee el tomador de la póliza con la compañía reduce el riesgo anual de no renovación en 6 % y un aumento de 1 año en la antigüedad del vehículo reduce en 3 % el riesgo anual de no renovación.

Tabla 3.7: Pruebas de significatividad del modelo de Cox multivariante

Prueba	Estadístico	Grados de libertad	Valor p
Likelihood ratio	743,5	4	<2E-16
Wald test	656,5	4	<2E-16
Score (logrank)	671,9	4	<2E-16

Fuente: Elaboración propia.

Por último, se realizó la validación del supuesto de proporcionalidad de los riesgos. Para esto, se hizo la prueba estadística de los residuos escalados de Schoenfeld, la cual valida para cada variable y en forma global (para todo el modelo) que dichos residuos son independientes del tiempo, lo cual se verifica a través de la correlación del conjunto correspondiente de residuos escalados con el tiempo. La hipótesis nula consiste en que se cumple el supuesto de proporcionalidad constante, es decir, que no hay correlación significativa entre los residuos y el tiempo. Los resultados de esta prueba pueden verse en la Tabla 3.8, a partir de los cuales se concluye que las variables *polholder_job* y *vehicl_age* no cumplen este supuesto y esto lleva a que el modelo en forma global tampoco cumpla el supuesto de riesgos proporcionales.

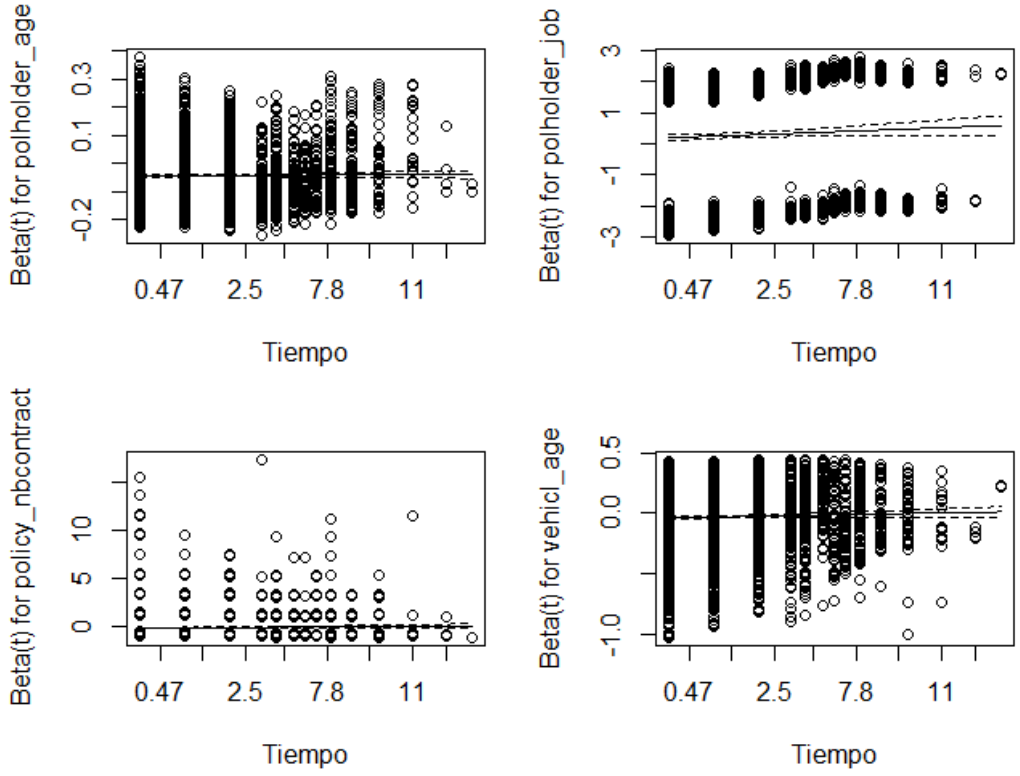
Tabla 3.8: Prueba estadística de los residuos escalados de Schoenfeld

Variable	Chi-cuadrado	Valor p
polholder_age	0,804	0,370
polholder_job	4,965	0,026
policy_nbcontract	3,050	0,081
vehicl_age	5,497	0,019
Global	13,649	0,009

Fuente: Elaboración propia.

Asimismo, se hizo la prueba gráfica de la distribución de estos residuos a través del tiempo, en este sentido, la prueba tiene como hipótesis nula que la recta de regresión lineal ajustada sobre estos residuos tiene pendiente cero, es decir, que los residuos se distribuyen aleatoriamente alrededor de una línea horizontal sin relacionarse con el tiempo. En la Figura 3.6 se muestran los resultados y se constatan las conclusiones de la prueba estadística, ya que para esas mismas variables se observa una recta con pendiente diferente de cero y ligeramente positiva. Dado que las variables empleadas y el modelo en forma global resultaron significativos pero se cumple parcialmente el supuesto de proporcionalidad, los resultados de este modelo deben tomarse con cautela.

Figura 3.6: Prueba gráfica de los residuos escalados de Schoenfeld



Fuente: Elaboración propia.

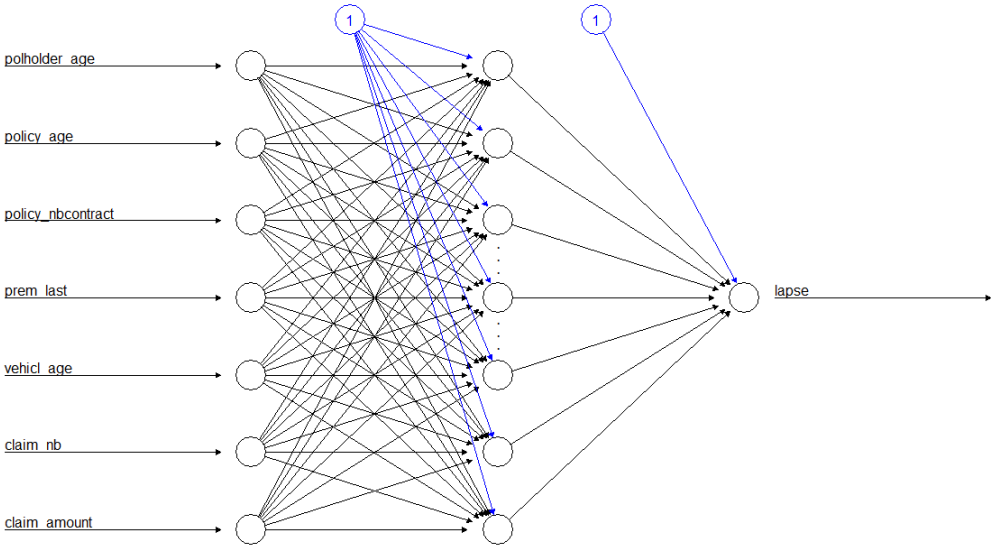
En la industria aseguradora, este tipo de análisis puede servir para entender diferentes comportamientos de la cartera, no sólo el tiempo de vida de la póliza sino también del cliente, el tiempo hasta el primer siniestro o entender, a través de un modelo multivariante, cómo afectan positiva o negativamente diferentes variables descriptivas del cliente o de la póliza a la probabilidad o al riesgo de cualquier característica que se modelice con esta técnica.

3.5. Modelos de clasificación: redes neuronales artificiales, árboles de decisión y máquinas de vectores soporte

3.5.1. Redes neuronales artificiales

Las redes neuronales artificiales se usaron para predecir la deserción de los clientes, siendo entonces un problema de clasificación. Para poner a prueba la efectividad de las redes neuronales artificiales como metodología propuesta, esta clasificación se abordó a través de otros métodos de aprendizaje automático (árboles de decisión y máquinas de vectores soporte) y poder ver en forma comparativa el desempeño de diferentes métodos de clasificación para predecir la deserción de los clientes. La red neuronal artificial modelizada puede verse en la Figura 3.7, en donde se tiene la capa de entrada con 7 entradas (7 variables predictoras), la capa oculta con 80 neuronas (vista simplificada en la Figura 3.7) y la capa de salida con la variable respuesta *lapse*. Dada esta configuración, la red neuronal tuvo un total de 721 pesos.

Figura 3.7: Red neuronal artificial modelizada

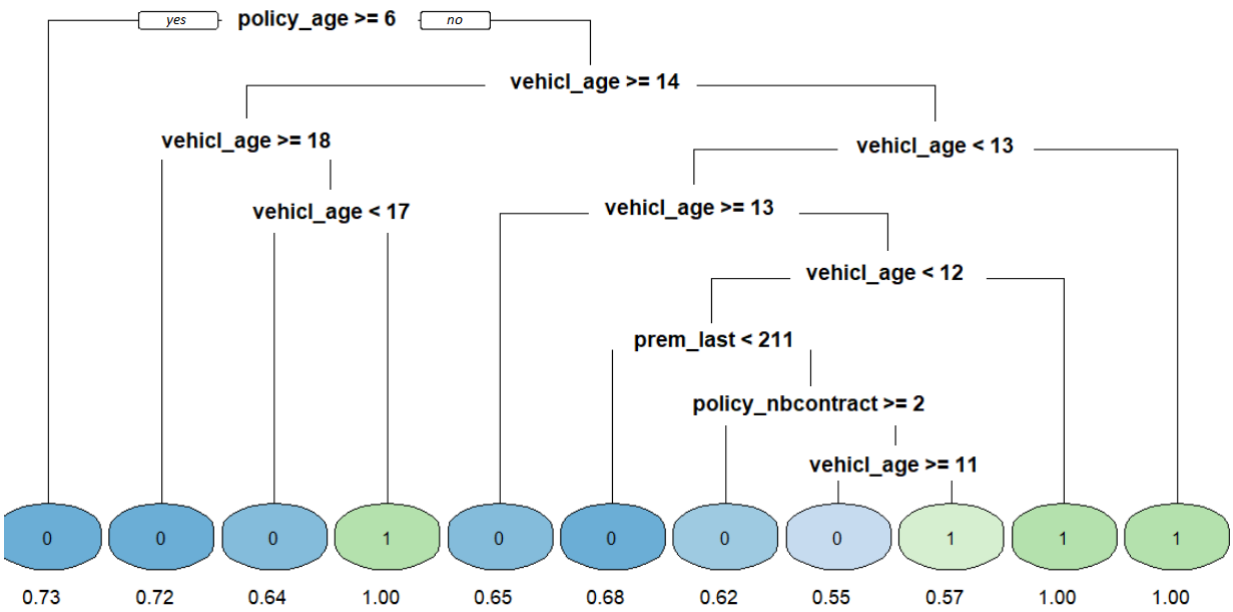


Fuente: Elaboración propia.

3.5.2. Árboles de decisión

En la Figura 3.8 se aprecia el árbol de decisión generado, el cual está conformado por un conjunto de 11 reglas de decisión, las cuales corresponden a los nodos terminales de dicho árbol. Para entender un poco su resultado y analizando, por ejemplo, el nodo terminal más a la derecha de la Figura 3.8, se tiene que si la antigüedad de la póliza es menor de 6 años y si la antigüedad del vehículo está entre 13 y 14 años, entonces la póliza no se renovará con una probabilidad de 1.

Figura 3.8: Árbol de decisión modelizado



Fuente: Elaboración propia.

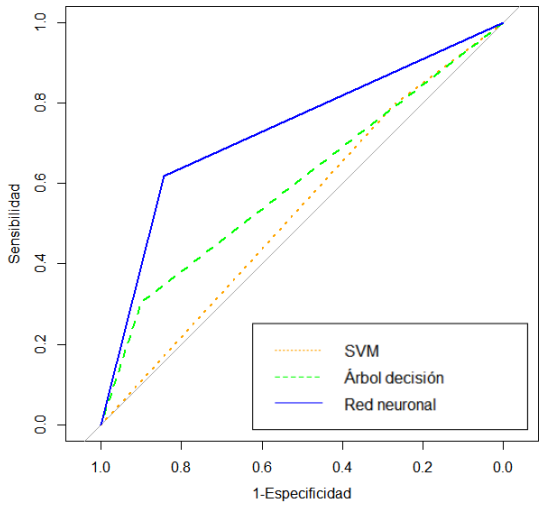
3.5.3. Máquinas de vectores soporte

Por su parte, el modelo de máquinas de vectores soporte generó un total de 2082 vectores de soporte, los cuales corresponden con las observaciones que tocan las líneas que definen el margen máximo del hiperplano de separación generado; esas observaciones constituyen puntos de contacto que se llaman vectores porque son vectores en el espacio p-dimensional, donde p es el número de variables.

3.5.4. Comparación de técnicas de clasificación

Teniendo en cuenta los diferentes modelos de clasificación y a partir de la Figura 3.9 y de la Tabla 3.9, se concluye que las redes neuronales son el modelo que presentó los mejores resultados, con una exactitud de 75 %, una sensibilidad (fracción correcta de verdaderos positivos) y una especificidad (fracción correcta de verdaderos negativos) mayor respecto a los otros modelos, así como también generó una mayor área bajo la curva (*area under curve* -AUC-). Por otro lado, el modelo de árboles de decisión evidencia una especificidad similar al modelo de máquinas de vectores soporte pero su exactitud, sensibilidad y AUC son superiores a los de éste. Las máquinas de vectores soporte tienen la más baja exactitud y sensibilidad y por tanto es el modelo con un menor desempeño; sólo destacan por su especificidad lo cual puede verse por la acumulación de área hacia la derecha según la forma que tiene su correspondiente curva ROC (*Receiver Operating Characteristic*, una representación gráfica del desempeño del clasificador).

Figura 3.9: Curvas ROC de los modelos de clasificación



Fuente: Elaboración propia.

Tabla 3.9: Métricas de desempeño de los modelos de clasificación

Métrica	Red neuronal	Árbol decisión	SVM
Exactitud	0,75	0,66	0,48
Especificidad	0,77	0,66	0,67
Sensibilidad	0,72	0,67	0,42
AUC	0,73	0,60	0,53

Fuente: Elaboración propia.

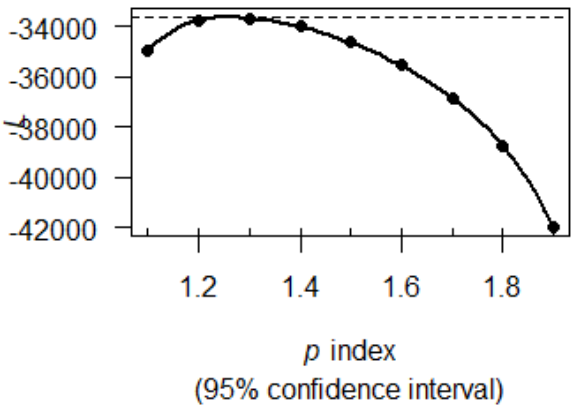
La implementación de este modelo de redes neuronales puede ayudar a predecir los clientes

que potencialmente abandonarán el negocio (las pólizas que no se renovarán) y a tomar acciones preventivas contra dicha deserción como alguna estrategia de fidelización del cliente o políticas de descuento en la próxima renovación, incluso puede medirse el grado de propensión a la deserción ya que entre los resultados de las redes neuronales artificiales, el modelo entrega también la probabilidad de pertenecer a la categoría objetivo, en este caso, la deserción.

3.6. Modelos de tarificación: Ridge, Lasso, ElasticNet y xGBoost

En esta sección se emplearon distintas técnicas para abordar el objetivo de tarificación de un seguro de coche. Primero se presentan los resultados de cada técnica y posteriormente se presentan los resultados comparativos. El enfoque de tarificación consiste en asignar el precio adecuado para cada póliza (la prima pura) según las características del tomador, de la póliza y del vehículo asegurado, por tanto, la prima corresponde con la pérdida esperada, es decir, la esperanza matemática de la siniestralidad, la cual es función de un conjunto de variables explicativas. Estas características son los factores de riesgo. Considerando que se empleó la distribución Tweedie para modelizar el comportamiento de la variable dependiente, en este caso, la prima pura, el primer paso consistió en determinar el parámetro p de la distribución. Recuérdese que la distribución Tweedie es una distribución Poisson compuesta Gamma y que el parámetro p de esta distribución se encuentra en el intervalo (1, 2), de modo que si $p=1$ la distribución Tweedie es una Poisson y si $p=2$ la distribución Tweedie es una Gamma. Para esto se realizó la estimación de p a través de máxima verosimilitud y se encontró que $p=1{,}263265$, como puede verse en la Figura 3.10. Con este resultado se realizaron los diferentes modelos de regresión Tweedie. Como función de enlace del GLM se usó una función logarítmica, lo cual generó un modelo multiplicativo con coeficientes relativos al nivel base.

Figura 3.10: Estimación del parámetro p de la distribución Tweedie



Fuente: Elaboración propia

3.6.1. GLM Tweedie

Para tener un escenario base con el cual se pudieran comparar los resultados de las demás técnicas propuestas para tarificación, se realizó una regresión Tweedie a través de GLM, lo cual corresponde con el enfoque tradicional para tarificación. En la Tabla 3.10 puede verse el análisis de la varianza (ANOVA) para este modelo GLM, en donde puede concluirse que las variables *polholder_gender* y *policy_nbcontract* no resultaron significativas para el modelo.

Tabla 3.10: ANOVA del GLM Tweedie

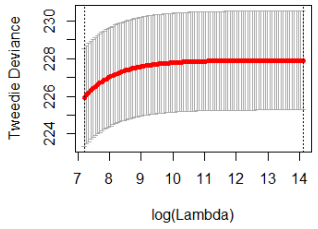
Variable	Chi-cuadrado	Grados libertad	Valor p
polholder_age	1684,80	2	<2E-16
polholder_BMCevol	12604,50	2	<2E-16
polholder_diffdriver	9413,10	6	<2E-16
polholder_gender	3,20	1	0,07277
polholder_job	6409,00	1	<2E-16
policy_age	580,30	2	<2E-16
policy_caruse	2220,90	2	<2E-16
policy_nbcontract	5,90	2	0,05188
prem_freqperyear	7495,30	3	<2E-16
vehicl_age	3558,60	2	<2E-16
vehicl_garage	24269,40	7	<2E-16
vehicl_powerkw	2861,70	10	<2E-16
vehicl_region	15594,00	13	<2E-16

Fuente: Elaboración propia.

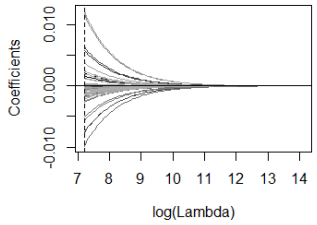
3.6.2. Regularizaciones Ridge, Lasso y ElasticNet

Estas regularizaciones requirieron estimar el parámetro de penalización λ, y adicionalmente, ElasticNet requirió estimar el parámetro de ajuste α; estas estimaciones se efectuaron a través del método de validación cruzada. Los resultados de las iteraciones de la validación cruzada pueden verse en la Figura 3.11.

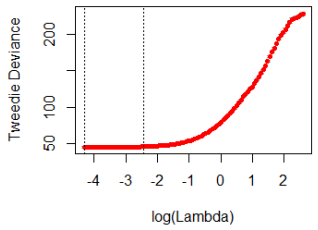
Figura 3.11: Estimación del parámetro de penalización λ y del parámetro de ajuste α



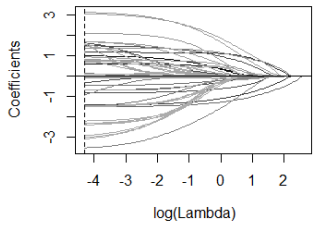
(a) Devianza de Ridge según λ



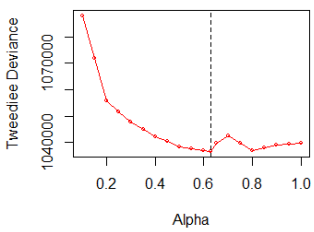
(b) Contracción de coeficientes de Ridge según λ



(c) Devianza de Lasso según λ



(d) Contracción de coeficientes de Lasso según λ



(e) Devianza de ElasticNet según α

Fuente: Elaboración propia.

Se obtuvieron los parámetros que minimizaron la devianza de cada modelo: Ridge ($\lambda = 1344,919$), Lasso ($\lambda = 0,01344919$) y ElasticNet ($\alpha = 0,63$ y $\lambda = 0,02647433$). Cuanto mayor sea la magnitud del parámetro de penalización, más se contraerán los coeficientes pudiendo llegar a hacerse cero, como muestran los paneles (b) y (d) la Figura 3.11. Estas técnicas de regularización se emplearon para la regresión Tweedie. En los paneles (a) y (c) de la Figura 3.11, también puede verse cómo varía la devianza en función de λ, pues el método de validación cruzada permite obtener el λ que haga menor la medida de error. Recuérdese que ElasticNet es una combinación de Ridge y Lasso. Tanto para la regularización Lasso como para ElasticNet, se usaron las variantes de dichas técnicas que permiten seleccionar grupos de variables (*group Lasso* y *group ElasticNet*), ya que en este caso todas las variables son categóricas y su matriz de diseño así lo requiere ya que en dicha matriz los diferentes niveles de una misma variable aparecen codificados como variables binarias adicionales.

3.6.3. xGBoost

La técnica de Extreme Gradient Boosting, conocida como un meta-algoritmo, fue utilizada a través de dos tipos de *boosters*. Un *linear booster* que permite realizar regresiones lineales, pudiendo así usar la distribución Tweedie para la variable respuesta y que también permite incorporar parámetros de regularización para lo cual se usó el α y el λ de ElasticNet. Por otra parte, un *tree booster* que no está asociado a ninguna distribución estadística y lo que hace es construir un conjunto de árboles de decisión aplicable a variables continuas.

3.6.4. Comparación de técnicas de tarificación

A continuación se presentan los resultados comparativos de las diferentes técnicas de tarificación para poder sacar conclusiones sobre el desempeño de cada técnica. La Tabla 3.11 muestra en forma comparativa las relatividades obtenidas para las variables y sus niveles respecto al nivel base. Puede verse que la regularización Ridge presenta los coeficientes de menor magnitud puesto que ésta es la naturaleza de la técnica. Por su parte, las regularizaciones Lasso y ElasticNet presentan coeficientes (y por tanto relatividades) muy similares entre sí y a su vez similares a los del GLM Tweedie pero de magnitud ligeramente menor. Nótese que las relatividades que aparecen con valor de 0 ó 1 se deben a la precisión decimal presentada en la tabla. En términos generales, las características que más elevan la tarifa del seguro respecto a la prima base son: un vehículo con antigüedad entre 0 y 2 años, con potencia de 275kW y no usar garage privado sino estacionar en la calle.

Tabla 3.11: Relatividades obtenidas por cada modelo

Variable y nivel	GLM Tweedie	Ridge	Lasso	ElasticNet	xGboost (linear)
polholder_age[19,24]	2,7240	1,0017	2,6777	2,6388	1,7779
polholder_age(50,85]	1,7994	1,0021	1,7951	1,7929	1,5885
polholder_BMCevoldown	0,2354	0,9924	0,2368	0,2387	0,2895
polholder_BMCevolup	3,2034	1,0058	3,1928	3,1854	2,9868
polholder_diffdriverall drivers > 24	0,4916	0,9993	0,4946	0,4989	0,7327
polholder_diffdrivercommercial	0,8472	0,9999	0,7121	0,7574	1,0000
polholder_diffdriverlearner 17	6,0760	1,0001	5,1047	4,3673	1,0000
polholder_diffdriveronly partner	0,4371	0,9924	0,4369	0,4375	0,4986
polholder_diffdriverunknown	0,0000	1,0000	0,9785	0,9883	1,0000
polholder_diffdriveryoung drivers	4,4398	1,0063	4,3867	4,3318	3,6067
polholder_genderFemale	0,9696	0,9983	0,9674	0,9640	0,9984
polholder_jobmedical	0,2140	0,9902	0,2161	0,2181	0,2566
policy_age(2,8]	0,6204	0,9950	0,6235	0,6255	0,9038
policy_age(8,17]	0,2115	0,9973	0,2238	0,2341	0,7454
policy_carusecommercial	0,0000	1,0000	0,3802	0,6500	1,0000
policy_caruseunknown	0,0264	0,9943	0,0286	0,0337	0,0957
policy_nbcontract(2,5]	0,9474	0,9994	0,9495	0,9496	1,0000
policy_nbcontract(5,15]	1,2828	1,0000	1,1488	1,1250	1,0000
prem_freqperyear12 per year	0,9102	0,9985	0,9171	0,9211	1,0000
prem_freqperyear2 per year	1,0699	0,9981	1,0702	1,0711	1,0000
prem_freqperyear4 per year	3,9784	1,0118	3,9658	3,9555	3,6958
vehicl_age[0,2]	8,2100	1,0013	8,1536	7,9788	3,8783
vehicl_age(2,10]	2,1829	1,0025	2,1811	2,1747	1,8574
vehicl_garagecarport	22,0484	1,0038	21,2418	20,2472	8,3653
vehicl_garageother	5,5643	1,0000	5,3689	5,1232	2,2002
vehicl_garageparking deck	2,3384	0,9981	2,2503	2,1428	1,0000
vehicl_garageprivate estate	0,9097	0,9997	0,9340	1,0127	1,0000
vehicl_garagestreet	23,8392	1,0123	22,8526	21,6081	8,9990
vehicl_garageunderground garage	2,6349	0,9996	2,5430	2,4264	1,0000
vehicl_garageunknown	5,6387	0,9984	5,3487	4,9974	1,0000
vehicl_powerkw100 kW	0,9716	0,9997	0,9649	0,9596	1,0000
vehicl_powerkw125-300 kW	2,0663	1,0012	2,0401	2,0131	1,5258
vehicl_powerkw150 kW	1,9744	1,0004	1,9412	1,9049	1,0230
vehicl_powerkw175 kW	2,2409	1,0001	2,1677	2,0702	1,0000
vehicl_powerkw200 kW	2,5499	1,0000	2,0882	1,7752	1,0000
vehicl_powerkw225 kW	4,6326	1,0002	4,2750	3,8877	1,0000
vehicl_powerkw25-50 kW	0,4888	0,9974	0,4884	0,4893	0,5803
vehicl_powerkw250 kW	5,6914	1,0001	4,7270	3,8332	1,0000
vehicl_powerkw275 kW	28,6344	1,0000	5,5443	2,7032	1,0000
vehicl_powerkw300 kW	11,4220	1,0000	4,8401	2,7823	1,0000
vehicl_regionReg1	0,0426	0,9991	0,0479	0,0582	0,5245
vehicl_regionReg10	0,0852	0,9987	0,0889	0,0958	0,3897
vehicl_regionReg11	0,0430	0,9992	0,0481	0,0575	0,4919
vehicl_regionReg12	0,0912	0,9984	0,0942	0,1000	0,3339
vehicl_regionReg13	0,0477	0,9992	0,0530	0,0633	0,5153
vehicl_regionReg14	0,2237	0,9989	0,2276	0,2357	0,5911
vehicl_regionReg2	0,0390	0,9992	0,0447	0,0557	0,5561
vehicl_regionReg3	0,2228	0,9987	0,2270	0,2345	0,5273
vehicl_regionReg5	0,2117	0,9978	0,2172	0,2247	0,4808
vehicl_regionReg6	0,0909	0,9986	0,0952	0,1030	0,4244
vehicl_regionReg7	1,0539	1,0025	1,0634	1,0838	1,5430
vehicl_regionReg8	0,9544	1,0021	0,9686	0,9914	1,4402
vehicl_regionReg9	0,1052	0,9989	0,1081	0,1138	0,3442

Fuente: Elaboración propia.

Por otra parte, es importante evaluar el desempeño de cada modelo en términos de la precisión de las predicciones realizadas. La Tabla 3.12 presenta en forma comparativa varias métricas de error de pronóstico para cada modelo: la devianza (*deviance*), el error cuadrático medio (MSE), la raíz del error cuadrático medio (RMSE), el error absoluto medio (MAE) y el error porcentual absoluto medio (MAPE). A partir de estos resultados, se puede ver que el modelo con menor devianza es el GLM Tweedie; sin embargo los modelos Lasso, ElasticNet y xGBoost (linear) tienen devianza apenas un poco superior; por su parte, el modelo Ridge presenta una devianza cinco veces mayor y el concepto de devianza no es aplicable para el modelo xGBoost (tree). En cuanto al MSE y RMSE, los modelos GLM Tweedie y Lasso muestran el mayor error mientras que los modelos Ridge y xGBoost (linear) tienen un error menor y parecido entre sí, pero el modelo xGBoost (tree) presenta un error considerablemente menor. Y en cuanto al MAE y MAPE, el modelo Ridge es el que muestra peor desempeño; los modelos GLM Tweedie, Lasso y ElasticNet tienen un error similar entre sí; el modelo xGBoost (linear) se desempeña mejor que los anteriores y nuevamente el modelo xGBoost (tree) presenta un error mucho menor. En términos generales y en forma comparativa respecto a la técnica considerada como escenario base (GLM Tweedie), se obtuvo que con excepción de Ridge, las técnicas de regularización propuestas presentaron un mejor desempeño medido en error de pronóstico, lo cual tiene sentido porque precisamente éste es un objetivo de estas técnicas; por su parte, las técnicas de xGBoost, especialmente xGBoost (tree), presentaron un desempeño notoriamente mayor en comparación con la técnica tradicional de GLM para tarificación.

Tabla 3.12: Métricas de error de pronóstico

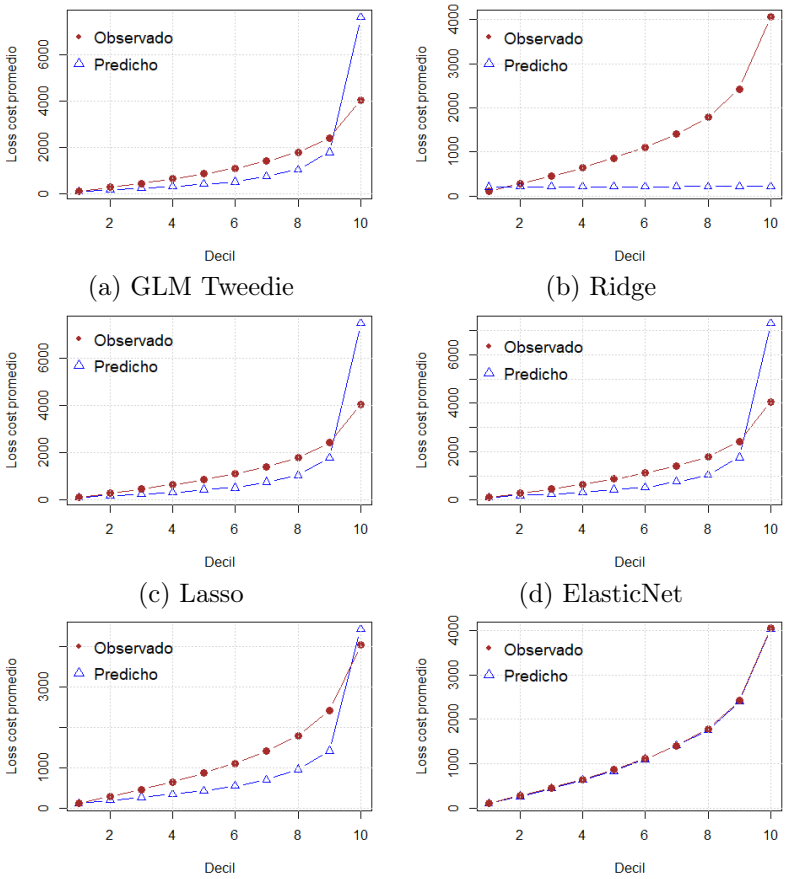
Métrica	GLM Tweedie	Ridge	Lasso	ElasticNet	xGboost (linear)	xGboost (tree)
Deviance	1003662,01	5209190,20	1005988,86	1010784,68	1436875,90	-
MSE	13034030,00	2626081,00	12375101,00	11457760,00	2658643,00	45395,22
RMSE	3610,27	1620,52	3517,83	3384,93	1630,53	213,06
MAE	858,11	1118,05	846,89	829,47	722,21	93,02
MAPE	0,7822	1,0325	0,7811	0,7785	0,8863	0,19

Fuente: Elaboración propia.

Adicionalmente, en el campo actuarial existen otras formas de comparar modelos de tarificación. Éstas tienen que ver con las medidas de *lift* que ayudan a entender la capacidad que tiene el modelo para prevenir la selección adversa, y por tanto, permitir otorgar a cada asegurado una tarifa justa según su nivel de riesgo, logrando así una cartera heterogénea con subcarteras homogéneas entre sí. Como primera medida de *lift* se emplearon los gráficos de cuantiles, los cuales son una representación directa de la habilidad del modelo para diferenciar entre los mejores y los peores riesgos. En este caso se emplearon deciles. A partir de este gráfico, un buen modelo debe cumplir los siguientes criterios: primero, precisión predictiva, es decir, confrontar el promedio de cada cuantil predicho versus el observado; segundo, monotonicidad, es decir, que la prima pura predicha sea monótona creciente conforme aumentan los cuantiles; tercero, distancia vertical entre el primer y el último cuantil, esto es, que el primer cuantil contenga los mejores riesgos y que el último cuantil contenga los peores riesgos para la compañía, de manera que haya una clara distinción entre estos riesgos. La Figura 3.12 muestra los gráficos de cuantiles para los diferentes modelos. Se puede observar que el modelo Ridge presentó una tarifa plana, casi equivalente para todos los deciles y que su precisión predictiva es muy mala a partir del cuarto decil. Los modelos GLM Tweedie, Lasso y ElasticNet tienen gráficos muy semejantes y todos ellos cumplen los criterios de precisión (con excepción del decil 10, en donde predicen primas muy elevadas respecto a las primas observadas en ese decil), monotonicidad y distancia vertical, por lo que puede decirse que distinguen bien los riesgos. Por su parte, los modelos xGBoost cumplen también los tres criterios mencionados, con la

particularidad de que consiguen predicciones más ajustadas a los valores observados y en especial, el modelo xGBoost (tree) logra realizar predicciones muy cercanas a los valores observados, hecho confirmado por las bajas métricas de error presentadas anteriormente. El modelo xGBoost (tree) es el que cumple de la mejor forma los tres criterios que se esperan visualizar en el gráfico de cuantiles.

Figura 3.12: Gráficos de cuantiles



(a) GLM Tweedie (b) Ridge

(c) Lasso (d) ElasticNet

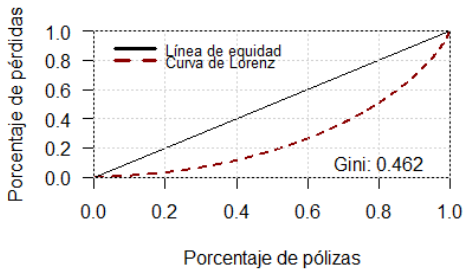
(e) xGBoost (linear) (f) xGBoost (tree)

Fuente: Elaboración propia.

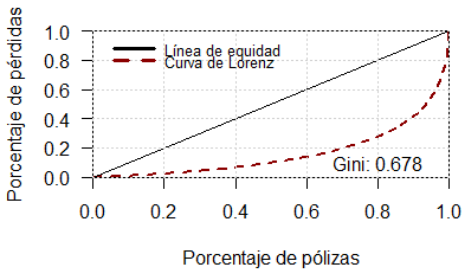
Por último, como medida de *lift* adicional se emplearon los coeficientes de Gini y las curvas de Lorenz. El coeficiente de Gini sirve para medir una distribución con desigualdad y toma valores entre 0 (igualdad perfecta) y 1 (desigualdad perfecta). Ambos vienen de un concepto económico para cuantificar la inequidad de un país pero también ayudan a segmentar la cartera entre los mejores y los peores riesgos. La curva de Lorenz muestra en términos porcentuales cómo se distribuye la pérdida esperada acumulada respecto a la población acumulada de pólizas. En esta curva, la línea de 45 grados es llamada *línea de igualdad*, según la cual, si cada asegurado representara la misma pérdida para la compañía, entonces la curva de Lorenz estaría sobre la línea de igualdad y la cartera tendría un coeficiente de Gini de 0 ya que no habría diferencia entre grupos de riesgo. La Figura 3.13 muestra las curvas de Lorenz de los datos originales y de cada modelo generado. En general, puede verse que el 60 % de los asegurados producen aproximadamente el 20 % de las pérdidas por siniestros. En coherencia con los resultados de los gráficos de cuantiles presentados anteriormente, puede verse que el modelo de Ridge no es capaz de reconocer diferentes riesgos, por lo cual su coeficiente de Gini es casi 0 y su curva de Lorenz casi coincide con la línea de igualdad. Los modelos GLM Tweedie, Lasso y ElasticNet son los que mejor logran reconocer heterogeneidad

en los riesgos. Por su parte, el modelo xGBoost (tree), pese a que tiene el menor coeficiente de Gini entre los modelos propuestos, tiene un coeficiente de Gini muy similar a los datos observados, dado su preciso poder predictivo respecto a los datos observados.

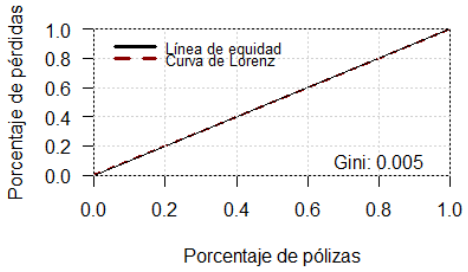
Figura 3.13: Curvas de Lorenz y coeficientes de Gini



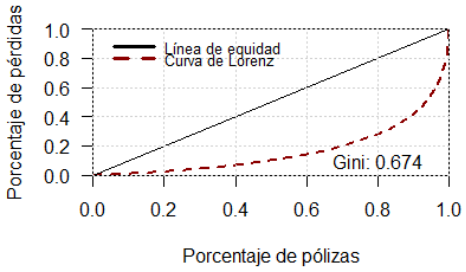
(a) Datos observados



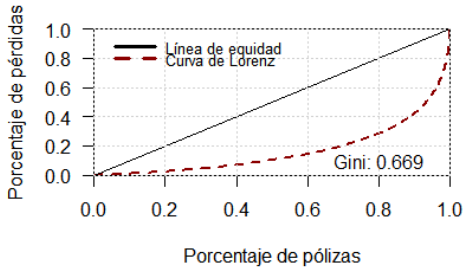
(b) GLM Tweedie



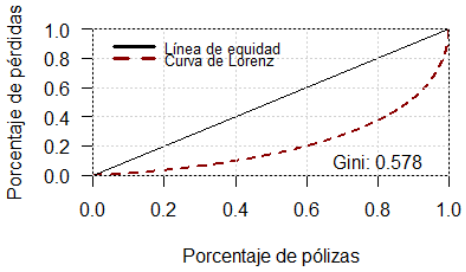
(c) Ridge



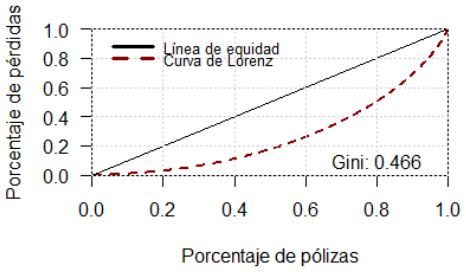
(d) Lasso



(e) ElasticNet



(f) xGBoost (linear)



(g) xGBoost (tree)

Fuente: Elaboración propia.

Recapitulando, estas técnicas de tarificación sirven para asignar la prima pura adecuada a cada asegurado según sus características de riesgo y conforme realicen una acertada distinción de los diferentes niveles de riesgo, generarán primas justas eliminando la selección adversa. Esto se traduce en ofrecer precios competitivos en el mercado a la vez que la empresa mantendrá sus costes de siniestralidad y su rentabilidad controlada, ya que por la Ley de los Grandes Números, la siniestralidad particular de cada asegurado es difícil de pronosticar, pero cuanto mayor sea el tamaño de la cartera y más precisa la predicción de la siniestralidad, más acertada será la estimación de la pérdida total de la cartera.

Conclusiones

En este trabajo se implementaron varios modelos de aprendizaje automático, algunos de ellos para entender y predecir el comportamiento de los clientes y otros para realizar tarificaciones que siguieran los criterios de equidad con los asegurados, discriminación de riesgos y suficiencia para la empresa. En ambos casos, los resultados fueron satisfactorios. El primer grupo de modelos se enfocó en demostrar que a través de estas técnicas se puede obtener conocimiento valioso y estratégico. El segundo grupo de modelos permitió demostrar que existen técnicas alternativas que pueden presentar resultados competitivos o incluso mejores en algunos aspectos, en comparación con las técnicas usadas tradicionalmente.

En relación con las técnicas que se utilizaron para resolver la misma tarea, se obtuvo que en los modelos de clasificación para predecir la deserción del cliente, las redes neuronales artificiales lograron unos resultados superiores en comparación con los árboles de decisión y las máquinas de vectores soporte. Por otra parte, en los modelos de tarificación se obtuvo que el algoritmo xGBoost logró precisiones más altas en comparación con los demás modelos propuestos, e incluso, mayores en comparación con la metodología empleada en la actualidad de modelos lineales generalizados. Sin embargo, se encontró que los modelos bajo enfoques actuales tampoco presentan mal desempeño, no obstante se pueden mejorar. En este sentido, podrían fusionarse modelos creando así modelos híbridos que potencien las características de cada modelo por separado.

Por otra parte, ciertos modelos de aprendizaje automático pueden ser de especial utilidad para superar inconvenientes de los supuestos de linealidad exigidos en algunos modelos, o de la presencia de distribuciones subyacentes necesarias en los modelos paramétricos. Sin embargo, a pesar de las ventajas ofrecidas por algunos de estos nuevos modelos, dentro del sector puede haber una barrera de interpretabilidad, porque sin importar la exactitud entregada por los modelos, algunos de ellos, como las redes neuronales artificiales, denominadas modelos de «caja negra», no dejan claro el funcionamiento de la toma de decisiones del algoritmo y por tanto, tampoco el efecto de las variables predictoras sobre la variable respuesta, lo cual puede ser una desventaja si se quieren tener modelos que ayuden a entender la relación entre estas variables. De todas formas cabe mencionar que no todos los modelos de aprendizaje automático tienen esta limitación de interpretabilidad de sus resultados.

Además, la ciencia actuarial no es ajena al entendimiento y uso de las técnicas de aprendizaje automático, es más, la profesión actuarial tiene todas las capacidades para adoptar estas nuevas técnicas que eventualmente pueden convertirse en herramientas enormemente útiles. Incluso, algunos softwares actuariales ya están incluyendo modelos de aprendizaje automático dentro de sus paquetes.

Para terminar, las aplicaciones de los modelos de aprendizaje pueden ser tan variadas como los retos del día a día de una compañía de seguros. Por ejemplo, ya pueden emplearse modelos de aprendizaje automático para detectar el fraude en la declaración de siniestros; para determinar las reservas de los siniestros ocurridos pero no declarados, típicamente trabajados mediante mé-

todos como Chain Ladder; e incluso, dentro del campo del aprendizaje automático denominado «aprendizaje profundo», ya se están utilizando algoritmos de análisis de imagen que permiten hacer peritación remota de los siniestros. Otros enfoques modernos combinan, por ejemplo, técnicas de aprendizaje automático con lógica difusa, existiendo así las redes neuronales difusas. Es decir, el potencial de estas técnicas es muy grande y se mantiene en constante desarrollo. En definitiva, los modelos de aprendizaje automático tienen una amplia y provechosa aplicabilidad en la industria aseguradora.

Bibliografía

Adankon, M., & Cheriet, M. (2007). Optimizing resources in model selection for support vector machine. *Pattern recognition, 40*(3), 953-963.

Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules. In *Proceedings of the 20th international conference on Very large data bases (VLDB), 1215*, 487-499.

Bischl, B., Lang, M., Bossek, J., Horn, D., Richter, J., & Surmann, D. (2017). *BBmisc: Miscellaneous Helper Functions for B. Bischl.* R package version 1.11.

Bishop, C. (2006). *Pattern recognition and machine learning.* Cambridge, United Kingdom: Springer.

Biswamohan, D., & Bidhubhusan, M. (2012). E-CRM practices and customer satisfaction in insurance sector. *Research Journal of Management Sciences, 1*(1), 2-6.

Charpentier, A. (2014). *Computational Actuarial Science with R.* New York, United States of America: Chapman and Hall/CRC.

Charpentier, A., & Dutang, M. (2019). *Package 'CASdatasets'.* R package version 1.0-10.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining,* 785-794.

Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., Chen, K., Mitchell, R., Cano, I., Zhou, T., Li, M., Xie, J., Lin, M., Geng, Y., & Li, Y. (2019). *xgboost: Extreme Gradient Boosting.* R package version 0.82.1.

Cox, D. (1972). Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological), 34*(2), 187-202.

Cristianini, N., & Shawe-Taylor, J. (2001). *An introduction to support vector machines and other kernel-based learning methods.* Cambridge, United Kingdom: Cambridge University Press.

Dunn, P. (2017). *Tweedie: Evaluation of Tweedie exponential family models.* R package version 2.3.

Flórez, R., & Fernández, J. (2008). *Las redes neuronales artificiales.* La Coruña, España: Netbiblo.

Fox, J., & Weisberg, S. (2019). *An R Companion to Applied Regression.* Thousand Oaks, United States of America: Sage.

Friedman, J. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics, 29*(5), 1189-1232.

Goldburd, M., Khare, A., & Tevet, D. (2016). Generalized linear models for insurance rating. *Casualty Actuarial Society, CAS Monographs Series*, (5).

Guillén, M., Nielsen, J., Scheike, T., & Pérez-Marín, A. (2012). Time-varying effects in the analysis of customer loyalty: A case study in insurance. *Expert Systems with Applications, 39*(3), 3551-3558.

Hahsler, M., Buchta, C., Gruen, B., & Hornik, K. (2019). *arules: Mining Association Rules and Frequent Itemsets*. R package version 1.6-4.

Hoerl, A., & Kennard, R. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics, 12*(1), 55-67.

Huang, Z. (1997). Clustering large data sets with mixed numeric and categorical values. *In Proceedings of the First Pacific Asia Knowledge Discovery and Data Mining Conference*, Singapore: World Scientific, 21–34.

Jørgensen, B., & De Souza, M. (1994). Fitting Tweedie's compound Poisson model to insurance claims data. *Scandinavian Actuarial Journal*, (1), 69–93.

Kaplan, E., & Meier, P. (1958). Nonparametric estimation from incomplete observations. *Journal of the American statistical association, 53*(282), 457-481.

Kasman, A., & Turgutlu, E. (2011). Performance of European insurance firms in the single insurance market. *International Review of Applied Economics, 25*(3), 363-378.

Kassambara, A., Kosinski, M., & Biecek, P. (2019). *survminer: Drawing Survival Curves using ggplot2*. R package version 0.4.6.

Kenneth, H., & Gentleman, R. (2019). *muhaz: Hazard Function Estimation in Survival Analysis*. R package version 1.2.6.1.

Kleinbaum, D., & Klein, M. (2010). *Survival analysis*. New York, United States of America: Springer.

Kuhn., M. (2020). *caret: Classification and Regression Training*. R package version 6.0-86.

Larose, D. (2014). *Discovering knowledge in data: an introduction to data mining*. New Jersey, United States of America: John Wiley & Sons.

Leinweber, D. (1979). *Models, complexity, and error*. Santa Monica, United States of America: RAND Corporation.

Loshin, D., & Reifer, A. (2013). *Using information to develop a culture of customer centricity: customer centricity, analytics, and information utilization*. Waltham, United States of America: Elsevier.

Matis, C., & Ilies, L. (2014). Customer relationship management in the insurance industry. *Procedia Economics and Finance, 15*(14), 1138-1145.

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., & Leisch, F. (2019). *e1071: Misc Functions of the Department of Statistics, Probability Theory Group* (Formerly: E1071), TU Wien. R package version 1.7-3.

Milborrow, S. (2019). *rpart.plot: Plot rpart Models: An Enhanced Version of plot.rpart*. R package version 3.0.8.

Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in neurorobotics, 7*, 21.

Ngai, E. (2005). Customer relationship management research (1992-2002). *Marketing intelligence & planning, 23*(6), 582-605.

Olabe, X. (1998). *Redes neuronales artificiales y sus aplicaciones*. Bilbao, España: Publicaciones de la Escuela de Ingenieros.

Parvatiyar, A., & Sheth, J. (2001). Customer relationship management: Emerging practice, process, and discipline. *Journal of Economic and Social Research, 3*(2), 1-34.

Pol, A. (1993). Modelo de regresión de Cox: ejemplo numérico del proceso de estimación de parámetros. *Psicothema, 5*(2), 387-402.

R Core Team (2019). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Raorane, A., Kulkarni, R., & Jitkar, B. (2012). Association rule-extracting knowledge using market basket analysis. *Research Journal of Recent Sciences, 1*(2), 19-27.

Richards, K., & Jones, E. (2008). Customer relationship management: Finding value drivers. *Industrial marketing management, 37*(2), 120-130.

Richeldi, M., & Perrucci, A. (2002). Churn analysis case study. *Deliverable, 2*(17), 1-12.

Riquelme, J., Ruiz, R., & Gilbert, K. (2006). Minería de datos: Conceptos y tendencias. *Inteligencia Artificial: Revista Iberoamericana de Inteligencia Artificial, 10*(29), 11-18.

Robin, X., Turck, N., Hainard, A., Tiberti, N., Lisacek, F., Sanchez., J, & Müller., M. (2011). pROC: an open-source package for R and S+ to analyze and compare ROC curves. *BMC Bioinformatics, 12*(1), 77.

Russell, S., & Norvig, P. (2016). *Artificial intelligence: a modern approach*. Malaysia: Pearson Education Limited.

Santana, Ó. (1991). El análisis de cluster: aplicación, interpretación y validación. *Papers: revista de sociología*, (37), 65-76.

Schelldorfer, J., & Wuthrich, M. (2019). *Nesting classical actuarial models into neural networks*. Social Science Research Network ID 3320525.

Siber, R. (1997). Combating the churn phenomenon. *Telecommunications, 31*(10), 77–81.

Smyth, G., & Jørgensen, B. (2002). Fitting Tweedie's compound Poisson model to insurance claims data: dispersion modelling. *ASTIN Bulletin: The Journal of the IAA, 32*(1), 143-157.

Szepannek, G. (2018). clustMixType: User-Friendly Clustering of Mixed-Type Data in R. *The R Journal*, *10*(2), 200-208.

The Sales Educators (2006). *Strategic sales leadership: BREAKthrough thinking for BREAKthrough results*. Mason, United States of America: Thomson.

Therneau, T. (2020). *A Package for Survival Analysis in R*. R package version 3.1-11.

Therneau, T., & Atkinson, B. (2019). *rpart: Recursive Partitioning and Regression Trees.* R package version 4.1-15.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological), 58*(1), 267-288.

Torgo, L. (2010). *Data Mining with R, learning with case studies.* Porto, Portugal: Chapman and Hall/CRC Press.

Tsiptsis, K., & Chorianopoulos, A. (2011). *Data mining techniques in CRM: inside customer segmentation.* Chichester, United Kingdom: John Wiley & Sons.

Venables, W., & Ripley, B. (2002). *Modern Applied Statistics with S.* New York, United States of America: Springer.

Wei, Q., Yi, Y. & Hui, Z. (2013). *HDtweedie: The Lasso for the Tweedie's Compound Poisson Model Using an IRLS-BMD Algorithm.* R package version 1.1.

Whiteson, S., Tanner, B., Taylor, M., & Stone, P. (2011). Protecting against evaluation overfitting in empirical reinforcement learning. In *2011 IEEE symposium on adaptive dynamic programming and reinforcement learning (ADPRL)*, Paris, 120-127.

Yuan, M., & Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 68*(1), 49-67.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67*(2), 301-320.

Apéndice A

Análisis descriptivo de los datos

A.1. Transacciones de compra

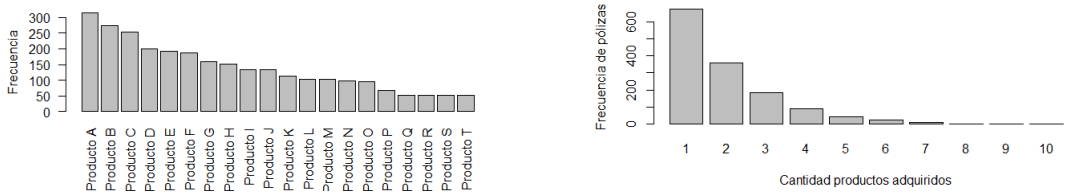
Los datos de *Transacciones de compra* están formados por 1390 pólizas y una variedad de 20 paquetes flexibles disponibles denominados como *Producto A, Producto B, . . . , Producto T* comprados por estas pólizas. Cada póliza puede adquirir uno o varios de estos paquetes. Un ejemplo de los datos puede verse en la Tabla A.1. Por su parte, en la Figura A.1(a) puede verse la frecuencia de compra de cada producto. La nomenclatura de los productos obedece a la frecuencia de compra en forma descendente, es decir, el producto más comprado es el Producto A, seguido del Producto B, etc. Además, se tiene que de las 1390 pólizas, 717 pólizas (correspondiente a un 52 % del total) compran más de un producto, por lo que el Análisis de la cesta de compra tiene sentido sobre este subconjunto de pólizas. Por otro lado, sólo una póliza compró 10 productos, siendo ésta la mayor combinación de varios productos en una única póliza dentro de este conjunto de datos. En la Figura A.1(b) puede verse esta distribución según la cantidad de productos.

Tabla A.1: Ejemplo de los datos *Transacciones de compra*

policy	product
B00120001	Producto M
B00120001	Producto B
B00120001	Producto A
B00120001	Producto E
B00120001	Producto F
B00120002	Producto D
B00120002	Producto H
B00120003	Producto R
B00120003	Producto A
B00120004	Producto N
...	...

Fuente: Elaboración propia.

Figura A.1: Distribución de los datos de *Transacciones de compra*



(a) Frecuencia de compra por producto (b) Frecuencia de pólizas según cantidad de productos

Fuente: Elaboración propia.

A.2. European lapse dataset from the direct channel

Los datos de *European lapse dataset from the direct channel* están formados por 23060 registros y 21 campos, contando con 11 variables cuantitativas y 10 variables cualitativas. Un ejemplo de los datos puede verse en la Tabla A.2.

Tabla A.2: Ejemplo de los datos *European lapse dataset from the direct channel*

lapse	polholder_age	polholder_BMCevol	polholder_diffdriver	polholder_gender	polholder_job	policy_age	...
0	38	stable	only partner	Male	normal	1	...
1	35	stable	same	Male	normal	1	...
1	29	stable	same	Male	normal	0	...
0	33	down	same	Female	medical	2	...
0	50	stable	same	Male	normal	8	...
0	37	stable	only partner	Male	normal	1	...
0	24	up	same	Female	medical	1	...
0	52	down	learner 17	Male	medical	1	...
0	32	down	same	Female	normal	1	...
0	80	stable	same	Male	normal	9	...
...	...	...	...	...	...	...	...

Fuente: Elaboración propia.

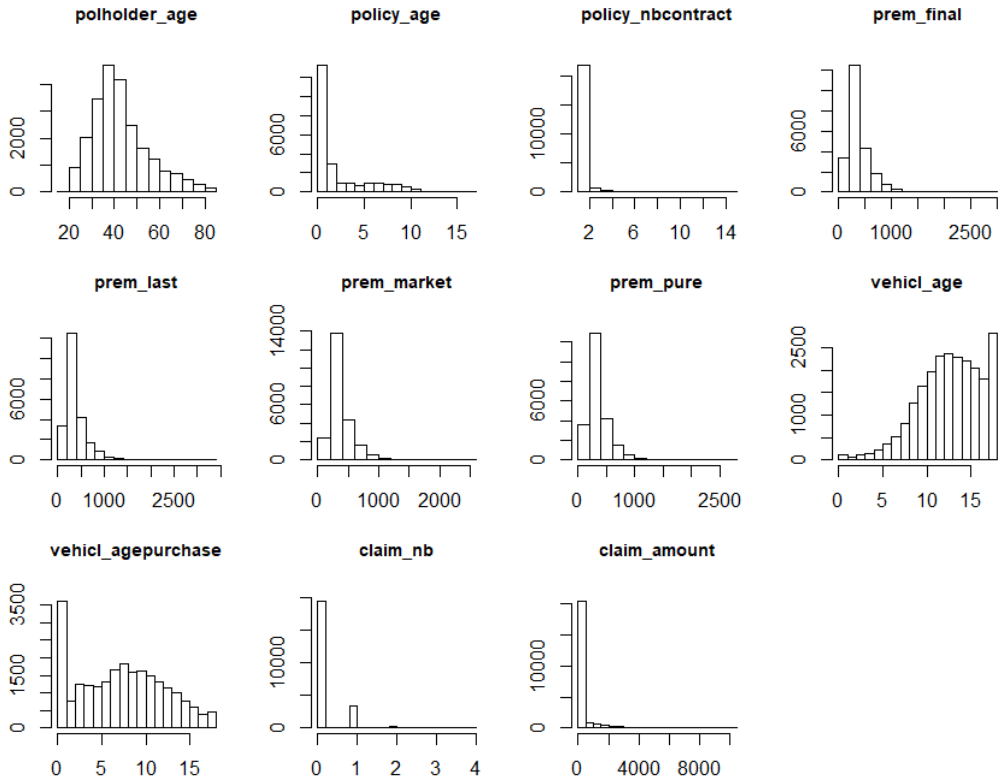
Adicionalmente, en la Tabla A.3 puede verse un resumen descriptivo de las variables cuantitativas y en la Figura A.2 la distribución de estas mismas variables.

Tabla A.3: Resumen estadístico de las variables cuantitativas

Variable	Mínimo	Cuartil 1	Mediana	Media	Cuartil 3	Máximo	Asimetría	Curtosis
polholder_age	19,00	35,00	41,00	43,05	49,00	85,00	0,89	3,64
policy_age	0,00	0,00	1,00	2,44	4,00	17,00	1,33	3,64
policy_nbcontract	1,00	1,00	1,00	1,31	1,00	15,00	5,49	57,65
prem_final	46,55	232,84	312,25	374,12	448,37	2948,05	2,24	12,18
prem_last	46,56	232,63	311,00	380,51	449,60	3362,07	2,24	11,25
prem_market	50,11	245,15	316,83	373,53	434,45	2416,84	2,41	12,58
prem_pure	45,55	227,10	301,44	355,88	423,56	2716,08	2,40	13,84
vehicl_age	0,00	11,00	13,00	13,06	16,00	18,00	-0,64	3,24
vehicl_agepurchase	0,00	4,00	8,00	7,68	11,00	18,00	0,06	2,08
claim_nb	0,00	0,00	0,00	0,18	0,00	4,00	2,39	8,82
claim_amount	0,00	0,00	0,00	214,70	0,00	100049,90	4,63	30,76

Fuente: Elaboración propia.

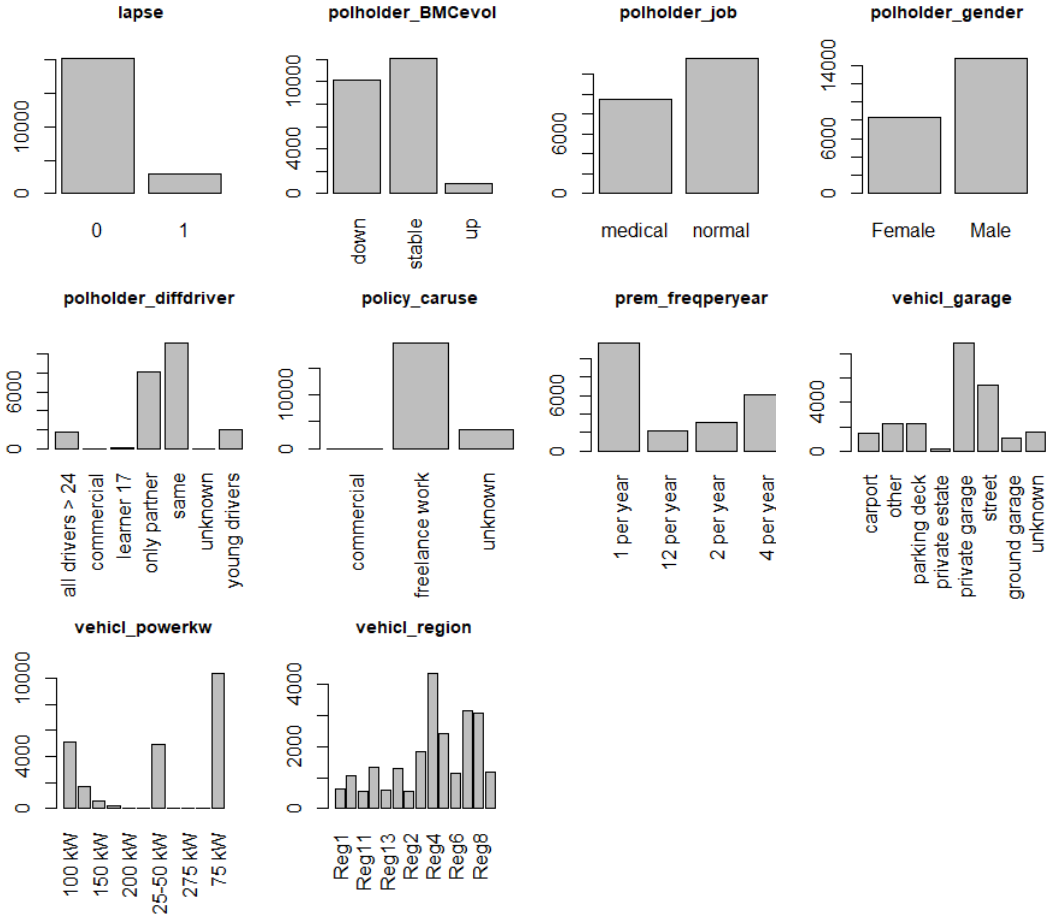
Figura A.2: Distribución de las variables cuantitativas de *European lapse dataset from the direct channel*



Fuente: Elaboración propia.

Asimismo, en la Figura A.3 se tienen las distribuciones de las variables cualitativas así como los diferentes niveles posibles en cada una de ellas.

Figura A.3: Distribución de las variables cualitativas de *European lapse dataset from the direct channel*



Fuente: Elaboración propia.

También se muestra en la Tabla A.4 los coeficientes de correlación de Pearson entre las variables cuantitativas. Puede verse una alta correlación positiva (muy cercana a 1) entre las variables *prem_final*, *prem_last*, *prem_market*, *prem_pure*, lo cual se explica porque todas esas variables hacen referencia al importe de los distintos tipos de prima asociados a la póliza. Existe otra correlación moderadamente fuerte entre *claim_nb* (número de siniestros) y *claim_amount* (cuantía de los siniestros) de 0,678. También hay una correlación de 0,598 entre *vehicl_age* y *vehicl_agepurchase* ya que ambas se refieren a antigüedades del vehículo. Las demás variables cuantitativas no indican correlaciones fuertes ni en sentido positivo ni negativo.

Tabla A.4: Coeficientes de correlaciones entre las variables cuantitativas

	polholder_age	policy_age	policy_nbcontract	prem_final	prem_last	prem_market	prem_pure	vehicl_age	vehicl_agepurchase	claim_nb	claim_amount
polholder_age	1,000	0,422	0,043	-0,256	-0,271	-0,277	-0,242	0,083	-0,271	0,002	0,002
policy_age		1,000	0,066	-0,160	-0,177	-0,324	-0,151	-0,030	-0,403	-0,002	0,002
policy_nbcontract			1,000	-0,048	-0,051	-0,063	-0,049	-0,008	-0,011	-0,002	-0,004
prem_final				1,000	0,951	0,895	0,991	-0,021	0,332	-0,003	-0,001
prem_last					1,000	0,839	0,926	-0,023	0,337	-0,004	-0,003
prem_market						1,000	0,899	0,008	0,384	0,000	0,000
prem_pure							1,000	-0,018	0,326	-0,002	-0,001
vehicl_age								1,000	0,598	-0,001	-0,010
vehicl_agepurchase									1,000	0,003	-0,005
claim_nb										1,000	0,678
claim_amount											1,000

Fuente: Elaboración propia.

Por su parte, la Tabla A.5 muestra los coeficientes de asociación V de Cramer para las variables cualitativas, indicando cierto grado de asociación entre *polholder_ diffdriver* y *policy_ caruse* (0,365) y entre *vehicl_ powerkw* y *vehicl_ region* (0,307).

Tabla A.5: Coeficientes de asociación entre las variables cualitativas

	lapse	polholder_BMCevol	polholder_diffdriver	polholder_gender	polholder_job	policy_caruse	prem_freqperyear	vehicl_garage	vehicl_powerkw	vehicl_region
lapse	1,000	0,088	0,039	0,019	0,019	0,056	0,041	0,031	0,018	0,063
polholder_BMCevol		1,000	0,063	0,056	0,023	0,129	0,150	0,098	0,098	0,036
polholder_diffdriver			1,000	0,047	0,108	0,365	0,079	0,095	0,067	0,041
polholder_gender				1,000	0,133	0,016	0,063	0,066	0,227	0,040
polholder_job					1,000	0,082	0,137	0,058	0,068	0,067
policy_caruse						1,000	0,111	0,242	0,044	0,051
prem_freqperyear							1,000	0,088	0,039	0,085
vehicl_garage								1,000	0,052	0,109
vehicl_powerkw									1,000	0,307
vehicl_region										1,000

Fuente: Elaboración propia.

Apéndice B

Paquetes R

En la Tabla B.1 se describen los diferentes paquetes de R empleados para la realización de este trabajo.

Tabla B.1: Paquetes R

Paquete	Descripción
arules	Infraestructura para representar, manipular y analizar datos transaccionales y patrones (ítems frecuentes y reglas de asociación)
Bbmisc	Funciones varias de ayuda para trabajar con datos. Incluye la función *normalize* para estandarizar datos
car	Funciones aplicadas a regresiones. Incluye la función *Anova* para calcular las tablas de análisis de la varianza
caret	Funciones misceláneas para entrenar y representar modelos de clasificación y regresión. Incluye la función *createDataPartition* para particionar conjuntos de datos
clustMixType	Funciones para realizar clustering k-prototypes para variables de tipo mixto
DMwR	Funciones y datos correspondientes al libro "Data Mining with R, learning with case studies" de Luis Torgo, CRC Press (2010). Incluye la función *SMOTE* para equilibrar conjuntos de datos no balanceados
e1071	Funciones para análisis de clasificación, transformada de Fourier de tiempo reducido, clustering difuso, máquinas de vectores soporte, computación del camino más corto, clasificador Naïve Bayes, entre otros
HDtweedie	Algoritmo *iteratively reweighted least square* (IRLS) que incorpora un método de *blockwise majorization descent* (BMD), para calcular de manera eficiente la solución de Lasso (agrupado) y de ElasticNet (agrupada) para el modelo Tweedie
muhaz	Estimación suavizada de la función de riesgo
nnet	Redes neuronales prealimentadas con una capa oculta y para modelos multinomiales log-lineales
pROC	Herramientas para visualizar, suavizar y comparar curvas ROC, áreas bajo la curva (AUC) e intervalos de confianza
rpart	Particionamiento recursivo para árboles de clasificación, regresión y supervivencia
rpart.plot	Gráficos para modelos generados con el paquete rpart
survival	Rutinas para el análisis de supervivencia, incluyendo curvas de Kaplan-Meier, modelos de Cox y modelos paramétricos del tiempo de falla acelerado
survminer	Herramientas para visualizar curvas de supervivencia y curvas ajustadas para el modelo de Cox
tweedie	Cálculos de máxima verosimilitud para la familia de distribuciones Tweedie
xgboost	Algoritmo Extreme Gradient Boosting

Fuente: Elaboración propia.

Apéndice C

Código R

En este apéndice se muestra el código R fundamental utilizado para elaborar cada uno de los modelos aplicados en este trabajo, por tanto, se prescinde del código relacionado con la carga de librerías, la lectura y preparación de datos, y la elaboración de tablas y gráficos básicos. Además, teniendo en cuenta que se presenta el código de cada modelo en una sección diferente, se propone la siguiente nomenclatura: *data* es el dataframe que contiene los datos para el modelo y *model* es el objeto que contiene los resultados del modelo.

C.1. Análisis de la cesta de compra

```
#Algoritmo a priori:
model = apriori(data, parameter = list(supp = 0.0025, conf = 0.8, maxlen =
    20))
#Eliminar reglas redundantes:
model.subset = which(colSums(is.subset(model, model)) > 1)
model = model[-model.subset]
#Ver reglas:
inspect(sort(model, by = "confidence"))
#Reglas para el Producto L:
l.model = apriori(data, parameter = list(supp = 0.002, conf = 0.6),
    appearance = list(default = "lhs", rhs = "Producto L"))
#Grafos del Producto L:
plot(l.model, method = "graph", engine = "htmlwidget")
```

C.2. Análisis clúster

```
#Método del codo:
k.max = 15
wss = sapply(1:k.max, function(k) {kproto(data, k)$tot.withinss})
plot(1:k.max, wss, type = "b", pch = 19, frame = FALSE, xlab = "Número de
    clusters k", ylab = "Suma total de cuadrados")
#Cluster k-prototypes con k=7:
model = kproto(data, 7, keep.data=TRUE)
```

C.3. Análisis de supervivencia

```
#Modelo de supervivencia Kaplan-Meier:
model = Surv(time = data$policy_age, event = data$lapse)
#Curvas Kaplan-Meier general (1) y por género (2):
fit1 = survfit(model~1, data = data)
fit2 = survfit(model~polholder_gender, data = data)
#Curvas acumuladas:
ggsurvplot(fit1, data = data, pval = TRUE, linetype = "strata", conf.int =
    TRUE, surv.median.line = "hv", fun = "event",xlab = "Tiempo (años)",
    ylab = "Probabilidad acumulada")
ggsurvplot(fit2, data = data, pval = FALSE, linetype = "strata", conf.int=
    FALSE, surv.median.line = "hv", fun = "event",xlab = "Tiempo (años)",
    ylab = "Probabilidad acumulada")
#Función de riesgo:
fit = muhaz(data$policy_age, data$lapse)
plot(fit, xlab = "Tiempo (años)", ylab = "Fuerza de mortalidad")
```

C.4. Regresión de Cox

```
#Modelos univariantes:
variables = c("polholder_age", "polholder_gender", "polholder_job", "
    policy_nbcontract", "prem_last", "vehicl_age", "claim_nb", "
    claim_amount")
formulas_univ = sapply(variables, function(x) as.formula (paste('Surv(time
    = data$policy_age, event = data$lapse)~', x)))
model_univ = lapply(formulas_univ, function(x) {coxph(x, data = data)})
#Modelo multivariantes:
model = coxph(Surv(time = data$policy_age, event = data$lapse) ~
    polholder_age + polholder_job + policy_nbcontract + vehicl_age, data =
    data)
#Validación proporcionalidad y residuos escalados de Schoenfeld:
res = cox.zph(model)
plot(res, df = 2, xlab = "Tiempo")
```

C.5. Modelos de clasificación: redes neuronales artificiales, árboles de decisión y máquinas de vectores soporte

```
#Red neuronal:
model = nnet(lapse ~ polholder_age + policy_age + policy_nbcontract +
    prem_last + vehicl_age + claim_nb + claim_amount, data = as.data.frame(
    data), size = 80, maxit = 100000)
#Arbol de decisión:
prune = rpart.control(minsplit = 4500, minbucket = 250)
model = rpart(lapse ~., data=data, control = prune)
#Máquinas de vectores soporte:
model = svm(formula = lapse ~., data = data, type = "nu-classification", nu
    =0.1, kernel = "linear")
```

C.6. Modelos de tarificación: Ridge, Lasso, ElasticNet y xGBoost

```
#Estimación del parámetro p de la distribución Tweedie:
p = tweedie.profile(claim_amount~., data = data, p.vec = seq(1, 2, 0.1), method = "series", phi.method = "mle", do.plot = TRUE, do.points = TRUE, verbose = 3)
p$p.max
#GLM Tweedie:
model = glm(claim_amount~., data = data, family = tweedie(var.power = p$p.max, link.power = 0))
Summary(model)
#Datos para regularizaciones:
x = model.matrix(claim_amount~., data)
y = as.numeric(data$claim_amount)
#Datos para xgboost
datos = xgb.DMatrix(data = x, label = y)
#Regresión Ridge:
lambda_ridge = cv.HDtweedie(x = x, y = y, p = p$p.max, nfolds = 10, alpha = 1e-5)
model = HDtweedie(x = x, y = y, p = p$p.max, alpha = 1e-5)
coef.HDtweedie(model, s = lambda_ridge$lambda.min)
#Regresión Lasso:
lambda_lasso = cv.HDtweedie(x = x, y = y, p = p$p.max, nfolds = 10, alpha = 1)
model = HDtweedie(x = x, y = y, p = p$p.max, alpha = 1)
coef.HDtweedie(model, s = lambda_lasso$lambda.min)
#ElasticNet:
model = HDtweedie(x = x, y = y, p = p$p.max, alpha = 0.63, lambda = 0.02647433)
coef.HDtweedie(model, s = 0.02647433)
#xGBoost (tree Booster):
params = list(booster = "gbtree", eval_metric = 'rmse', max_depth = 200, verbose = 2)
model = xgb.train(data = data, params = params, maximize = FALSE, watchlist = list(data), nrounds = 200)
#xGBoost (linear booster):
params = list(booster = "gblinear", alpha = 0.63, lambda = 0.02647433, objective = 'reg:tweedie', tweedie_variance_power = p$p.max, eval_metric = 'rmse', max_depth = 200, verbose = 2)
model = xgb.train(data = data, params = params, maximize = FALSE, watchlist = list(data), nrounds = 200)
extract.coef(model)
#Curva de Lorenz y coeficientes de Gini:
ineq(data,type = "Gini")
plot(Lc(data), col = "darkred", lwd = 2, lty = 2,xlab = "Porcentaje de pólizas", ylab = "Porcentaje de pérdidas"); grid()
```